PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : H04L 9/32 — A1

(11) International Publication Number: WO 00/56009

(43) International Publication Date: 21 September 2000 (21.09.00)

(21) International Application Number: PCT/US00/07174

(22) International Filing Date: 17 March 2000 (17.03.00)

(30) Priority Data:
09/270,874 17 March 1999 (17.03.99) US

(71) Applicant: NEWTON, Farrell [US/US]; 8 Brighton 10th Path, Brooklyn, NY 11235 (US).

(71)(72) Applicants and Inventors: WILLIAMS, Gareth [US/US]; 35–11 85th Street, Jackson Hts, NY 11372 (US). MOORE, Charles, E., II [US/US]; 80 Varick Street, New York, NY 10013 (US). NICHOLS, Christopher, M. [US/US]; 80 Varick Street, New York, NY 10013 (US).

(74) Agent: SCHWEITZER, Fritz, L., III; Schweitzer Comman Gross & Bondell LLP, 230 Park Avenue, New York, NY 10163 (US).

(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: INTERNET, INTRANET AND OTHER NETWORK COMMUNICATION SECURITY SYSTEMS UTILIZING ENTRANCE AND EXIT KEYS

(57) Abstract

A method of providing user identification and authentication using ultra long identification key codes and/or ultra large databases of identification key codes in a manner providing secure access to a remote computer terminal to a database or server transaction program stored on a host computer.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

INTERNET, INTRANET AND OTHER NETWORK COMMUNICATION SECURITY SYSTEMS UTILIZING ENTRANCE AND EXIT KEYS

CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation-in-part of co-pending application Serial No. 60/100,462 filed September 15, 1998; which is a continuation-in-part of appliction Serial No. 09/037,297 filed March 9, 1998; which is a continuation-in-part of application Serial No. 08/570,318 filed December 11, 1995, now U.S. Patent No. 5,771,291.

BACKGROUND OF THE INVENTION

Most security programs for personal computers and networks rely upon simple user passwords and they are therefore vulnerable. There are two common methods for acquiring unauthorized access to a host computer. In the first method, the intruder improperly obtains and illegally uses the user ID and password of a valid user. The second method is to steal a valid user session in progress by switching the connection of the user to the thief's terminal. Without a method to verify the identity of the user, there is

little preventing an intruder from obtaining unauthorized access to the user's account through a purloined user ID and password.

This lack of security has been a shortcoming of various corporate and other networks including the Internet and is one factor that has limited commercial use of these networks.

One existing authentication system proposes to add a card reader to personal computers so that users can verify their identity with a user identification card, as shown in U.S. Patent 4,438,824, issued on March 27, 1984, to C. Mueller-Schloer for an invention entitled "Apparatus and Method for Cryptographic Identity Verification". However, few users will spend the time and money to install an expensive card-reader. Furthermore, user identification cards have very limited storage and usually store a short identification key. Therefore, the same short identification key is used during most if not all authentications.

United States Patent 5,371,792, entitled CD-ROM DISK AND SECURITY CHECK METHOD FOR THE

SAME issued on December 6, 1994 to Toshinori Asai and Masaki Kawahori, relates to CD-ROMs for television game devices. The purpose of the security check is to prevent unlicensed CD-ROM disks from being played on a Sega game machine. The CD-ROM disk identifier disclosed in this patent is not unique to each individual CD-ROM disk, but instead merely indicates a kind of the CD-ROM disk. All CD-ROM disks of the same type have the same disk identifier. In the patent, two kinds of identifiers, "SEGADISKSYSTEM" and "SEGABOOTDISC" are described. The security code indicates that the CD-ROM disk is duly licensed and also contains a program which generates a message displayed on the user's monitor that the disk is licensed.

There have been numerous patents issued for integrated circuit cards and other computerized portable security devices. For example, Beitel et al., U.S. Patent No. 4,430,728, employs a physical security key which is coupled into a connector provided for it at a remote terminal. The key has two access keys which are required to access the

central computer. This invention, like the Mueller-Schloer '824 credit card device, requires special hardware to be added to computers and requires costly security keys. Locking the terminal does not prevent intruders from procuring unauthorized access on public networks, since the intruder can use another terminal elsewhere.

SUMMARY OF THE INVENTION

The object of the present invention is to provide a practical and effective security system for secure remote terminal or terminal emulation or computer access to a host computer. This is accomplished by using ultra long passwords and/or ultra large databases of identification keys, i.e., by a CD-ROM disk or other portable large capacity storage medium containing a database of identification keys, long identification keys, or a combination thereof. The subsequent descriptions of the invention will be in terms of CD-ROM disks, although other portable storage media are contemplated for use, including Zip disks, floppy disks, digital versatile disks (DVD disks), Bernoulli disks, portable

hard drives (e.g. PCMCIA hard drives), and portable semiconductor memory units (e.g. PCMCIA memory units). The authentication system further includes a remote terminal with a portable large capacity storage medium reader or connector, and a communications device or system which connects the remote terminal to a host computer which has a large capacity storage medium.

A microprocessor or logic circuitry may be added to the portable memory medium in certain applications to implement additional security features or user features. Moreover the system of the present invention may be incorporated into a portable electronic devices.

In accordance with the invention, the new security system may utilize one or more CD-ROM disks, other portable storage media, other storage devices including redundant arrays of inexpensive disks and hard drives, or any hybrid thereof containing databases of the user identification keys.

The invention also contemplates encryption and other security methods for authenticating the identity of users.

Specifically, an enhanced security system entails the use of separate entrance and exit codes at the beginning and end respectively of the communication session, along with multiple authentication codes during each session, as required for super security. The invention also includes means for the tagging or identification of attackers who attempt to penetrate the new system; a programming means for such tagging or identification may be implemented on the portable storage media, in the central computers (servers) of the new system, or both.

DESCRIPTION OF THE DRAWINGS

Fig. 1 is a schematic diagram of an exemplary preferred embodiment illustrating the various steps required to practice the fundamental security system of the present invention, as well as illustrating the components which comprise the required hardware and software of one CD-ROM-based implementation of the fundamental system itself; and

Fig. 2 is a schematic diagram of an alternate preferred embodiment including a double-sided key or password.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS OF THE INVENTION

In general, the new and improved security system of the present invention provides individual users with what are characterized as "ultra long identification keys" which are embodied on a physical object such as a CD-ROM disk which is provided to the authorized individual user. By "ultra long" it is contemplated that the individual user code will comprise at least 20 characters or digits (requiring 20 or 10 bytes, respectively) of information as a bare minimum (it being understood that the typical password employed for consumer credit cards and the like is 16 characters), although the use of a CD-ROM disk "key" enables even passwords of hundreds of characters to be readily employed.

The following describes the use of CD-ROM disks as the portable storage medium; however, it is to be understood that the use of other portable storage media in lieu of CD-ROM disks is within the scope of this disclosure.

The initial step in the new security method is to generate individual user access codes for each and every

contemplated user who is to be granted authorized access to a network or a database or source or repository of information which is desired to be protected and which is stored in or in conjunction with a "home" server or base computer. The individual user access key codes are generated using algorithms or circuitry or combinations thereof which may be optionally provided with means to generate individual encryption keys as well, in accordance with well known methods and industry standards for generating encryption key codes. It is of course to be understood that in accordance with the principles of the present invention, the individual access key code is "ultra long" and is of a length that is otherwise too long and too cumbersome to be conveniently typed into a system by an individual.

A central registry or other compilation of all of the individualized user access codes is established and is optionally encrypted for loading on the home or main computer terminal or server on which the secured database is to be located or in association with which the server is to

function as a security mechanism. As a parallel to this step of the development of the security system, each of the individualized user access key codes is separately recorded, for example by ganged optical recording machines of the type known to the art for recording information onto CD-ROM disks. Each disk is in the form of a "CD-ROM key" which is individualized for a particular end user (for example, a customer of a catalog sales organization, a user of a secure database, a customer of a financial institution, etc.).

At this stage of the establishment of the system there is a complete registry of "ultra long" identification key codes stored in a server and there is a distribution of the physical CD-ROM disk keys to authorized individual users who are to be provided access to a database.

In order to provide authorized access to an authorized user of the database or "transaction program", the user at his remote personal computer terminal which is equipped with a CD-ROM reader, loads the CD-ROM disk into his computer and logs onto an access program or user

program (which may optionally be recorded on the CD-ROM disk as well). The user program then transmits the user's individual access key code (which optionally may be encrypted) over a communication network or over a telephone network to the host computer or server, which will be appropriately programmed to check the user's access key code against the registry of stored authorized individual user access key codes. The server program will further include the requisite steps to interdict and end any attempt to gain access to the server or transaction program through a transmitted access code which is not stored in the database of authorized individual user access key codes. The server program will disconnect and may optionally inform the user that an unauthorized key access code has been transmitted.

Alternatively, and assuming the CD-ROM disk was proper and contained an authorized access key code, the communication between the user's remote computer and the host server will continue with the host computer's program including steps to grant access to the user's program and

begin the session. As explained hereinafter, the host computer program or server program and the user program may optionally encrypt the session using the user's encryption key or keys, which are also stored in the server's database and on the individual user's CD-ROM disk. The optional encryption might also include encryption keys which are stored on the user's CD-ROM disk key.

At this stage, access to the secured database or "secured server transaction program" can proceed with the authorized user communicating through his own personal computer with the host server to conduct whatever "transaction" he may wish to effect, ranging from the simple ordering of merchandise, to the conduct of financial transactions, to conduct of research into a secured database, or any other type of two-way communication which is capable of being conducted between a remote computer terminal and a host computer over a communication network or a telephone network. It is to be understood that a level of security heretofore unavailable to remote consumers communicating with a host computer is provided

by the new system which utilizes ultra long identification key codes typically impressed upon or otherwise recorded upon "large keys" in the form of a CD-ROM disk or the like. The ultra long identification keys are checked and approved through databases of such identification keys which are stored in a remote host computer or server.

Security may also be enhanced by providing multiple keys or a database or table of keys (which may be a one-time pad of keys) on each user's CD-ROM disk.

The user program may provide the keys in sequence or according to a pre-arranged pattern or algorithm, or from a location requested by the server. The server might request the keys in sequence or from random locations; i.e. in a random order, or according to some other algorithm. It is important to note also that the user program may provide or the server may request more than one key or multiple keys at different times during the session. As hereinafter described, the use of a one-time pad of keys also insures that no key is transmitted twice; hence intercepting

or decrypting a key will not allow an attacker to gain access to the system.

In some applications, the key generation algorithm will run on the server itself or even on the users' computers; in the latter case, means to avoid generating duplicate keys is required (e.g. by a randomization function in the key generation algorithm or circuitry, plus a check for duplicate keys whenever a new key is added to the database).

Numerous other variants will also readily be apparent to those skilled in the art.

In a preferred embodiment, each user is issued a unique CD-ROM disk containing one or more unique identification keys. An individual user inserts his CD-ROM disk "key" into a computer connected via a network or other communications device to a host computer; also referred to herein as a server. An access program on the CD-ROM "key" connects to and forwards the unique identification key from the CD-ROM disk key to the host computer in encrypted form. A security authentication program stored on the server then decrypts the identification key, compares

the identification key with an identification key from the database of user identification keys located on a large capacity storage device connected to the host computer, and verifies the user's identity. The host computer or the user's access program may include a program or routine which will also demand that the user type in a password. If the identification key matches the identification key in the host computer's database of user identification keys and if the user enters the correct password, the host computer, through its programming, will grant access to the user.

The host computer (server) may be further programmed to require or challenge the user's remote access terminal program to re-authenticate itself at regular intervals or from time-to-time during the communication session. Or, the user's program may so reauthenticate itself and the host computer may be programmed to expect such reauthentication. This helps defend against attackers who try to capture an identification key en route to the host computer or who misappropriate or steal a user's connection. Unless an attacker has the user's unique CD-

ROM "key", he would be unable to use his unauthorized access for longer than the time between requested re-authentications. Means to insure that an intercepted identification key or message cannot be re-used by an attacker are discussed below.

Similarly, the user's program may require the host computer (server) to reauthenticate itself at regular intervals or from time-to-time during a communication session. Or, the server program may so re-authenticate itself and the user program may include code to expect such re-authentication. This helps defend against attackers who attempt to impersonate the host computer (server). Alternatively, the host computer and remotely accessed terminal program may request or expect identification keys periodically from each other.

It should be noted that such re-authentication may optionally be required at critical points in a communication session, e.g. to complete a transaction or to access a database. Such reauthentication may be required before, to initiate the action; or after, to validate the action; or both.

To insure that an intercepted identification key or message cannot be re-used by an attacker, defensive methods have been developed including the use of multiple, different identification keys; encrypting the identification keys or messages, ideally by time-dependent means, e.g. by combining the identification keys with time-of-day information, then encrypting.

Another defensive method for authenticating a user to the host computer and the host computer to the user with the identification keys is the exchange of identification keys one digit at a time. In a typical implementation of this method, the user's access program (running on the user's terminal or computer) transmits the first digit of its identification key to the host. The host computer determines whether the digit transmitted was correct. If the digit is correct, the host computer transmits the first digit of its identification key to the user's terminal or computer. The user's access program determines whether the digit returned by the host is correct. This process continues until either the user program and host computer have given each

other all the digits of their respective identification keys or until an incorrect digit is received by the host computer or user's access program.

Any attempt by an attacker to mimic either the host computer or the user terminal or computer most probably will fail on the first digit; if so, the attacker will get only one digit of the user password or host computer password. Thus, this technique provides additional security against "man in the middle" attacks aimed at illicitly obtaining a user or host password. Alternatively, several digits of the identification keys may be exchanged at each iteration, or single bits can be exchanged at each iteration, etc.

Although individual identification keys are contemplated, in some applications, some or all of these identification keys may be shared among a class or subclass of users.

In another embodiment, the host computer is programmed to send an encryption key to the remote terminal. The terminal program executing on the remote terminal uses the encryption key to encrypt the unique

identification key on the CD-ROM disk. Then the encrypted identification key is sent to the host computer for verification. If the encryption means is a public key encryption algorithm with a sufficiently long key, a third party would have great difficulty extracting the unencrypted identification. A variation to this method is to have part of the encryption key contained on the user's CD-ROM "key" with the other part sent from the host computer. The host computer always has access to a complete database of all the encryption keys and identification keys. Without the portion of the encryption key from the CD-ROM or host computer, the remote terminal program is unable to decrypt messages. If the encryption key from the host computer is varied with time, selected randomly, or unique to each user session, the user's computer will essentially never transmit the same encrypted identification key twice.

The remote terminal program may pad the identification key with random, null, or nonsense prefixes or suffixes or interpolated characters. To help insure that the same identification message is not sent twice, the encryption

algorithm is preferably provided with good diffusion (wherein a change in any character in the plain text changes many or all of the characters in the encrypted text). The pad will preferably be specified by the host computer so that previously used encrypted identification keys do not repeat.

The pad may vary in a pre-determined manner with time. For example, the pad may be the day, hour, and minute clock. The host computer will then be programmed to check that the pad is correct based upon the day, hour, and minute. The pad may also vary with each logon. Additionally, the user ID or user number may be padded as discussed above.

In another embodiment, the encryption key is included on the user's CD-ROM key disk and is never transmitted. The remote terminal program may pad the identification key as previously discussed. The host computer will be programmed to look up the encryption key for the user's claimed identity in a stored database of encryption and identification keys. Then the host computer

will decrypt the unique identification key, remove the padding, and compare the decrypted key with the key retrieved from the host computer database, thereby verifying the user's identity. Again, when the encryption algorithm has good diffusion, the added characters will insure that the user's computer will essentially never transmit the same identification key twice.

In another embodiment, the central server selects the encryption key of the moment from a table or database or pad of keys on the user's CD-ROM; a copy of the table being in the central server. This avoids transmitting the encryption key over the connection; all that is transmitted is which entry in the key table is to be used, not the encryption key itself; alternatively, the key may be selected by means known both to the server and user programs. These keys may also optionally be used to encrypt important information transmitted.

In another embodiment, the user's terminal program encrypts an authentication message, such as the user's identity, plus a varying padding, such as a random padding

or a predictably varying padding, such as the date and time, again using a key or encryption means unique to the user and stored on the user's CD-ROM or portable storage medium. The central server program looks up the appropriate key or encryption means for that user, decrypts the message, and checks the contents, thereby authenticating the user. A yet further alternative is for the user's terminal program to generate the authentication message by encrypting a predictably varying message, such as the time and date, again using encryption means unique to the individual user.

In another embodiment, the remote terminal transmits to the host computer a plain text or encrypted user ID or identification key from an identification key database on the user's CD-ROM key. A second encrypted identification key is sent from the remote terminal to the host computer. The first identification key is used by the host computer to look up a unique encryption key for that user. The second identification key is then decrypted using the unique encryption key and the user's claimed identity.

If the decrypted identification key is correct, the user's claimed identity is then verified. The encryption key is never transmitted since both the remote terminal and the host computer have the encryption key stored locally.

In addition, other parts of the transmission, or the entire transmission or session may be encrypted using a unique user-specific encryption key or keys on the user's CD-ROM disk. When the server is aware of the user's identity, it will look up the key in its own table: hence the key need never be transmitted between user and server or vice-versa. Again, techniques such as padding would typically be used. This embodiment not only provides additional security, it also is another way to authenticate the user's remote terminal program to the host computer. An "impostor" computer posing as the user terminal would lack the user's unique key or database of keys and would be unable to encrypt the user's messages to the host computer, and would be unable to decrypt the host computer's messages. In addition, this embodiment also securely authenticates the host computer to the remote terminal

program. An "imposter" server would lack the database of user encryption keys and would be unable to decrypt the remote terminal's messages and accordingly would be unable to respond plausibly to the remote terminal.

Alternatively, a one-time pad stored on both the user's CD-ROM disk key and the host computer may be used as the encryption means or key to encrypt the user's identification key to provide additional security. After receiving the encrypted identification key, the host computer is programmed to look up the one-time pad under the user's claimed identity in a database of one-time pads. After decrypting the identification key, the host computer will authenticate the user's identity.

Alternatively, a one-time pad of unique identification keys may be stored on each user's CD-ROM key disk. The central server would then demand a new key every time, and verify the new key against its own copy of that user's on-time pad of ID keys.

Both one-time pad arrangements also avoid transmitting the same user authentication key or message twice.

Furthermore, the one-time pad can be used to encrypt other important information communicated. For example, with use of a 250 kilobyte user-specific one-time pad (e.g. in conjunction with a consumer catalog) to encrypt the user's credit card number, assuming that one byte is used to encrypt each digit, then a sixteen digit credit card number would use 16 bytes of the 250 kilobyte one-time pad. Assuming the user performed ten transactions a day, the 250 kilobyte one-time pad would last more than four years. Note that, optionally, different one-time-pads may be used for identification keys and for encryption keys.

The central server can keep track of which one-time-pad keys have been used to prevent re-use. If the user's portable storage medium is writable, the user terminal software or access software may be used to keep a usage record or table or usage sequence number on the portable storage medium, or the user program may overwrite the

keys that have been used or set a flag bit or field associated with the keys that have been used. If the user only accesses the server from the one terminal, the user program may keep a usage record or table on the user terminal, e.g. on the hard drive.

Preferably, usage records may be kept in both the central server and on the user's portable storage medium or terminal, and any discrepancy between the usage records on the user's portable storage medium or terminal and on the server would suggest an attempt by a third party to illicitly gain access. Such a discrepancy will be indicated by any attempt by either the user program or the server to re-use a one-time-pad key or one-time-pad entry that has already been used with the server or user program respectively. Such a discrepancy will also be indicated by any attempt to use a key or pad entry out-of-sequence or any other "out-of-synch episode".

Thus, the server program (or user program) may assert that a particular key has been used (e.g. during an attempt by an imposter to gain access) even though the user

program (or server program) did not know that that key had been used. However, an attacker might try to mimic the server (or user) program, falsely claim that keys have been used, and thereby deplete the user s (or server s) one-time-pad of keys. To avoid this, the user (or server) program may optionally demand that the alleged server s program (or alleged user s program) provide one or more of the keys that it claims has been used. Alternatively or in addition, other authentication means as herein described or as are well-known to those skilled in the art may be used.

If the portable storage medium is not writable (as with conventional read-only CD-ROM disks), the user program cannot record on the portable storage medium which keys have been used. If only the central server keeps track of which keys have been used, an attacker might attempt to impersonate a server to the user program and request that the user program utilize a key that has already been used. There are several ways to eliminate this potential problem.

First, as previously mentioned, if the user only accesses the server from one terminal or PC or workstation, the user program may keep a usage record on the user terminal or PC or workstation; said record might typically be a small file or cookie or like stored on the hard disk drive. If the user accesses the server from only a few terminals or PCs or workstations (e.g. from a PC at the office and a PC at home), the user program can keep a separate usage record on each, and operate from different areas of the one-time-pad of keys, depending on which machine is being used. Thus, the user program would typically inform the server program which machine is being used, hence which area of the key pad is being used.

If the user utilizes his non-writable portable storage medium to access the server from any number of terminals or PCs or workstations, it eventually becomes impractical to allocate an area of the key pad to each. The user program cannot keep a record of which keys have been used on the nonwritable medium; therefore, it has no memory between sessions of which keys have been used.

However, it can keep track of which keys have been used during a session; e.g. it can keep a sequence number during a session in RAM. Thus, the user program must authenticate the server at the beginning of the session and concurrently get a correct sequence number or a pointer to an unused area of the key pad. (One way to do the latter is for the server to mathematically combine the sequence number with the authentication number or with a second authentication-type number, e.g. by addition. The user program independently calculates the authentication number and then obtains the sequence number, in this case by subtracting.) The main challenge is to authenticate the server; to do this, the user program typically generates a request to the server that differs at each sign-on and that requires the server to have a copy of the user's keypad. There are many ways to do this.

One of our techniques is to have a table of different initiation keys for different times. This is doable; for example, a separate one-time-pad of 20 digit (10 byte) initiation keys for each 5 minutes over a 3-year period

requires 315,360 keys or 3,153,600 bytes, which is less than 0.5% of a 650 Mbyte CD-ROM disk. Providing a different key for each minute for 3 years would require less than 2.5% of a 650 Mbyte CD-ROM disk. As an aside, the user program might optionally assert a clock time to the server; the server might optionally accept it if it is within a tolerance and does not correspond to a previously used time. Another option, if the time did not match, would be for the server to check the keys in the immediate neighborhood, again staying within a tolerance and rejecting any key that has been previously used.

Another technique for initial authentication of the server in the non-writable portable storage medium case is for the user program to generate a request for a key from a random location or address in the server's copy of the keypad. A single random number can be generated in any number of ways; one way is from the exact timing of one of the user's keystrokes. Alternatively, the user program can generate an address based on the time and date, or based on the process identification number (PID) for the program,

or from any number of other non-repeating numbers. In a preferred implementation, we set aside a separate table of initiation keys; and use a relatively small fraction of them, so as to minimize the chances of requesting a key that has been requested before. We derive an address in the key table from a random number, or we use a hashing function calculate an address in the table from the time-and-date or PID. We also allow a second or third (or a very limited number of) further choices if the server determines that a key has already been used. We may optionally further choose to have the user program request more than one key.

Yet another technique is for the user program to request a checksum or check-function calculated from a number of keys from different addresses in the server's copy of the keypad. Only the checksum or check-function is transmitted; the actual keys are not transmitted. The user program uses one or more non-repeating numbers as described in the previous paragraph to generate the addresses of the keys requested or to decide which keys to

request. In one preferred implementation, we again set aside a separate table of initiation keys; and use a relatively small fraction of them. The user program uses several random numbers derived from the exact timing of several of the user's keystrokes to select several 20-digit keys to request. The server sums those keys and forwards the lowest 20 bits of the sum to the user program. For an attacker to counterfeit this check-function, two conditions must be met. First, all of the keys must have been used before; if even one of them is new, and the keys are random, the result is a random number. Second, the new checksum must be calculable from the checksums in which the keys were previously used. The probability of both conditions being met can be made vanishingly small.

In addition, one may optionally use a check-function that depends upon the order of its arguments.

Note also that combinations or hybrids of the above techniques can be used. For example, a check-function may be calculated from a table entry from the current time-and-date, plus random or quasi-random keys from the

past; note that the present key has never been used and most of those past keys will also never have been used.

Yet further techniques for initial authentication of the server in the non-writable portable storage medium case will readily be apparent to those skilled in the art.

For any of the aforementioned user identification techniques, the terminal program and/or the host computer also may optionally be programmed to demand that the user enter by typing (through a keyboard) a password previously specified. The individual user s password may optionally be stored on the user s CD-ROM or other portable storage medium, in which case the terminal program compares said stored user password with the password entered by the user. Essentially, the user authenticates himself to the portable storage medium and the user terminal program, which in turn authenticate the portable storage medium to the host computer. Alternatively, the user s password may be stored on the host computer, and the host computer s program compares

the user s stored password with the password entered by the user.

All of the above-described encryption methods can also be used to encrypt important information transmitted.

All of the above-described authentication methods can also be used in reverse to authenticate the host computer to the remote terminal program, as will be readily understood.

The most secure encryption techniques, such as public key encryption, can take up to 1000 times longer to process than more routine encryption methods, unless a special-purpose processor for the particular algorithm is added to the user's computer. One method to increase speed is to use the most secure means to encrypt only the most sensitive portions of the transmission and use faster encryption methods for less critical portions of the transmission. Because of the large capacity and speed of a CD-ROM, databases or pads of encryption keys for each encryption method and host computer can be easily stored and accessed. Portions of the transmission that are common

and do not need to be protected can be transmitted as plain text. Repeated text or graphics which all users will view can optionally be stored on the CD-ROM to decrease the amount of information transmitted from the host computer to the remote terminal.

A special encryption device may be attached to the host computer in order to expedite encryption and decryption of transmitted data. Since the host computer will most likely service many users, the encryption device should prove economical when amortized over the large number of users.

The cost of having extremely large keys and databases of keys is the cost of the space on a CD-ROM which is not available for other information and the space needed to store these keys on the computer host. Since the cost of producing CD-ROM disks is modest, the use of CD-ROM disks has become quite economical. Thus the new authentication system of the invention is more economical and more effective than the prior art systems.

The cost of generating the keys is a lesser effect. Note that one-time-pad entries for either authentication or encryption can be generated by conventional algorithms, or by specialized hardware, such as a hardware random number generator; for the latter, one would typically use a random physical process, such as thermal noise or circuit noise or radioactive decay to generate the random bits or digits or numbers. We also contemplate using one-time-pad entries or hardware-generated random numbers as an input or seed to algorithms or hardware to generate keys for other encryption means. Note also that any key generation algorithm can be implemented in software or in hardware (e.g. a special-purpose processor), or in a mixture of the two; as some of our implementations require large numbers of keys for each user, we may optionally implement part or all of our key generation algorithms in hardware.

A yet further alternative is to generate or store keys at the central server and transmit them to the users' terminal programs only as they are needed, e.g. using one-time pads or other encryption means on the user's

portable storage medium; this avoids generating keys that may never be used. This is of particular utility when (for example) random numbers for one-time-pads are cheap and easy to generate, and the keys in question (e.g. using a product of two large prime numbers) is expensive or more difficult to generate.

Additionally, a user's CD-ROM key according to the invention may contain different identification keys or tables or databases of identification keys for use with different servers or to provide access to different databases or services on any individual server. For example, in an application wherein several catalogs of different vendors are contained on or accessed by one CD-ROM key, different databases of identification keys and encryption keys would be allocated to provide access to each vendor's host computer or database.

Also, a user's CD-ROM key according to this invention may contain different identification keys or tables or databases of identification keys to provide different levels of access to one or more host computers. Or, the host

computer may be programmed to grant different access privileges to different users or to different classes of users or different types of users. For example, in a corporate network, the president's CD-ROM key would grant maximum access to all information on the host computer, while a clerk's CD-ROM key would only grant limited access to specific data. Similarly, in a consumer application, different consumers might have different credit limits. The requisite privilege or privilege level might either be encoded on the CD-ROM or, preferably, would be included in a database on the host computer.

It will also be apparent that a single authorization server or set of authorization servers can be used to authorize access to many other servers or to may different databases or services. In this case, the table of what is authorized for a given user would typically be kept in the single server or set of servers for ease of updating, although it could be kept on the users' portable storage media (e.g. the user's CD-ROM disk) or on the central computer (server) or divided between the two.

If the user program contacts the other servers directly, the other servers can access the single server or set of servers to obtain the authorization; alternatively, the user program might contact the other servers through the single server or set of servers (e.g. if the authorization server function is implemented by an Internet service provider's server).

If a single server or set of servers is used to authorize access to other servers or to different databases or servers, new servers or databases or services can be authorized for a given user by simply updating the table of what is authorized for the user. Typically, the table or the portion of the table being updated would be in the single server or set of servers. However, if the user's portable storage medium is writable, an authorization table on the portable storage medium could be updated in the same fashion. Thus, the user will be able to use an existing CD-ROM or other portable storage medium to access new servers, databases or services.

It is also desirable to allow existing CD-ROM keys to be used to access new servers or databases or servers when the different host computers authorize access, as an alternative to or in lieu of referring access requests to a central server or set of servers per above. To do so, each CD-ROM disk would include identification keys or tables or databases of identification keys that are initially not assigned to any server or database or service. These would then be assigned later to access new servers, computers, programs, databases or information functions or services. This arrangement averts the need for distributing new CD-ROM disks whenever a new server is added.

Information about the new server or database or service, such as its name,, network address, and telephone number, along with the identification of the database of keys on the CD-ROM disk assigned to the new server must be added to the user's access program. For example, if 200 keys or key tables or one-time-pads of keys are already assigned to existing servers, the 201st key might be assigned to a new server. This information would be included (in

either encrypted or unencrypted form) on an update floppy disk or other portable medium, posted on a bulletin board or server, or updated automatically by the remote terminal access program during a subsequent communication session. Such information is typically the same for all users being granted access to the new server. If the user key is on a writable portable storage medium, the update information would typically be written directly to the portable storage medium.

If the portable storage medium is not writable, as with a conventional CD-ROM disk, the user's access program would typically store the update information for the new servers in a small file on the user's hard drives. If the users have a writable CD-ROM drive, the information could be added to the CD-ROM disk key. If the information about each server comprises no more than 50 characters, a 10 kilobyte disk file could contain information on at least 100 new servers. A file a few megabytes in size would allow a short description of each server.

Eventually, the new servers would be included on updated CD-ROM disk keys distributed to all users.

Informational, transactional, and promotional databases and services are all of ever-increasing commercial interest. Access can be controlled, verified, or tabulated by the CD-ROM key of the invention. In addition, the individual CD-ROM disks may be provided with all or portions of these databases. The portions of the databases that change infrequently might be encoded on the users' CD-ROM disks and updated when new disks are produced, whereas variable portions might typically be stored on the server.

The response speed of the user authentication system may be increased if the server or host computer being accessed begins the communication session in parallel with checking the user identification key from the user program against the database of user identification keys to authorize the user.. This may be advantageous if the database of keys has a slow response time, e.g. during peak usage hours. It may also be advantageous if the server or host computer

being accessed must take the time to contact another server or set of servers to check the database and obtain authorization, as discussed hereinbefore.

In such a case, it may be advantageous for the host computer being accessed to run a fast key-check algorithm to check whether the user identification key is a valid key, and whether or not it belongs to the particular user. In some applications, the server being accessed could use this validity check and then grant a provisional or limited access, pending checking of the user identification key against the database.

In addition, in certain applications, provisional initiation of the transaction upon receipt of a valid identification by the host computer might be permitted, but the transaction is completed only when the ID is verified in the server's database. This arrangement further improves response time for the user and reduces the speed requirements on the storage means. For example, a credit card transaction could be started upon receipt of a valid ID

but not completed until after the ID has been checked with the database and approved.

In one such key-check technique, the CD-ROM key of the invention may contain both unencrypted and encrypted versions of one or more identification keys. The encryption is done before or as the disk is imprinted using a key and encryption method unknown to the user and using encryption means that are ideally unknown to the user. For user authentication purposes, the host computer, which has the key, would be programmed to demand both the unencrypted version of the identification key and the encrypted version of the key. The host computer then would be programmed to decrypt the encrypted version of the key and compare it with the unencrypted version. If the two keys are the same, then the user identification key is virtually certainly a valid key. For example, if the encryption were the inverse of a long-key public-key encryption, the public key would be held by the host computer only (and the inverse or private key would be held by the disk maker only). An intruder would have to

generate a counterfeit identification with the corresponding encrypted version, which would require the inverse or private key. Obtaining the key would be virtually impossible, even if the would-be counterfeiter obtained huge numbers of different user disks. Since the server does not have the private key, even illicitly accessing the server would not allow a counterfeiter to make new counterfeit user identification keys. Accordingly, counterfeiting of valid user identification numbers cannot be done.

A further security measure includes appending the encrypted version of the identification key to the unencrypted version to form a single longer key. Alternatively, the final key may comprise two different encrypted versions of the unencrypted key. Alternatively, the final key may be a function of both the unencrypted version and of a parity, hash, encryption function, or other function of the unencrypted version.

Such key-validity-check algorithms help protect against attempts to counterfeit or simulate user disks or portable storage media; they do not protect against the use

of stolen user disks or portable storage media to gain at least provisional or limited access to the server.

One method to help protect against the use of stolen user disks or portable storage media to gain at least provisional or limited access in the above system is to provide each server with a list or database of known stolen keys; this database is much smaller than the complete database of user keys; it also can be checked more rapidly.

Unlike a human user, the computer does not make mistakes in entering an identification key. Accordingly, unless line disruption is indicated, the preferred software implementation will disconnect the user after only one attempt using any invalid CD-ROM identification key. This allows speedy rejection of attempts by attackers or other transgressors and avoids tying up the system with their illicit attempts. By disconnecting after one attempt, attackers cannot rapidly try multiple identification keys.

The host computer's database of user identification keys is well protected against attempts to steal or copy it. Nevertheless, it is advantageous to protect against attempts

to steal or copy the server's database of user identification keys or user access keys and thereby counterfeit or mimic the users' unique CD-ROMs. Accordingly, the server database of a preferred implementation of the invention contains an encrypted or otherwise altered version of the user identification keys. The server of the invention employs a trap-door authentication algorithm to compare the user ID or access key recovered from the incoming data stream with the altered version in the server's own database for that user's claimed identity. The trap-door authentication algorithm authenticates the user if and only if the encrypted identification key in the server's database represents the same identification key as the one embedded or encrypted in the incoming data stream. The trap-door authentication algorithm is impractical to be used to recover the actual identification key from the encrypted key in the host computer's database. Since the server database does not contain the actual identification keys, and the trap-door authentication function is of no help in recovering them, mere possession of the host computer's database is not

sufficient to recover the identification keys. Thus, stealing or copying the host computer's database of identification keys will not allow a thief to counterfeit the users' unique CD-ROM key access disks and thus will not allow the thief to access the system as a legitimate user.

One such trap door authentication algorithm is implemented as follows. When preparing the users' CD-ROMs and the database for the host computer, the users' unique identification keys are encrypted with a difficult-to-decrypt long-key code. The encrypted key is copied into the host computer's database and the unencrypted identification key is written onto the user's CD-ROM key. In use, the host computer takes the identification key recovered from the incoming data stream from the user, encrypts it with the same means used to encrypt the database, and compares the encrypted key with the database entry for that user. If the keys are identical, the user is authenticated and access is granted.

Another class of trapdoor authentication algorithms go directly from the encrypted version of the password

embedded in the data stream from the user to the other encrypted version in the server's database. Accordingly, the unencrypted version of the password never exists on the server and cannot be tapped or recorded by any illicit program or virus on the server.

In a yet further embodiment, each CD-ROM key is provided with multiple databases of identification and encryption keys. The server or host computer is programmed to use or have access only to one database. The copies of the other databases on the user's CD-ROM are stored in a vault. If the host computer's identification keys were ever stolen, the host computer can simply be loaded with one of the user databases from the vault and use the new identification keys. Since the user already has the new database of his new keys on his CD-ROM, there is no need to provide a new CD-ROM to all the users, and the thief remains locked out of the host computer. In addition, if only part of the server's database is copied or stolen, then only a portion of the database need be changed

and only the corresponding users' CD-ROM disks need use an alternative identification database.

In one implementation, the server then simply requests the new or different keys from the users' program rather then requesting the previously used keys: the users' programs access a different location on the users' CD-ROM keys or portable storage medium keys. If the users have individual databases or one-time pads of keys, the users' programs then access a different database on the users' CD-ROM keys or portable storage medium keys. The server might also transmit a re-authentication code to access any key or database of keys or one-time-pad of keys.

Preferably, a secure means to direct the users' computers to use a different database of identification keys on the CD-ROM is used. Any of the previously described authentication algorithms can be used for this purpose. One technique is for the server to encrypt by private key the message with a time-dependent pad. The user program on the CD-ROM then uses the public key, which is also stored on the CD-ROM, to decrypt the message, then checks that

the time-dependent pad is correct and switches to an alternate user ID or identification key database. The private key and the replacement database are given to the host computer at the same time.

The host computer may be provided with multiple databases wherein a specific combination is required to access any identification keys. For example, in one embodiment, one database contains a one-time pad and the other contains the database of identification keys encrypted using the one-time pad. A thief who stole or copied only the database would be unable to recover any keys.

In corporate applications, where the user CD-ROM keys will be used only or primarily on the company's own computers, the change to another user ID can be made permanent by recording a word in a small file on the hard drive. Once the file is altered on all of the company's computers, the change is complete. This could be done at the next log-on for each user.

In yet a further implementation, the host computer can use an array of inexpensive CD-ROM drives to store

the database of identification keys. Advantages of this novel CD-ROM array approach include that the cost per megabyte is comparable to or less than that of magnetic disk drives, and that a drive failure almost always leaves the recorded data intact. The CD-ROM disk can simply be changed to another drive. In addition, there is the security advantage that the written data is in permanent form.

As an occasional delay in a transaction is tolerable, magnetic tape can optionally be used as a back-up means or as a redundant storage means for use in regenerating data, or to store user keys or portions of the users' key tables or databases that are not yet needed. The storage means then comprises a fast storage means (e.g. CD-ROM disks or hard disk drives) that stores data that is apt to be needed in the near future, and a slow storage means with larger capacity and lower cost (the magnetic tape) to store keys that are not yet needed.

The users' CD-ROM disks may also contain a network access program, encryption routines, and other data and programs of utility to the users.

The portable large storage media may contain a read-only portion and a read-write portion, typically a write-once read-many portion or a write few, read many portion. (For the case of CD-ROM disks with writable portions, see, for example, the disks illustrated and described in U.S. patents 5,287,335 and 5,206,063, the disclosure of which is incorporated by reference herein.) The read-only portion would typically contain programs or information common to many users, e.g. network access programs and/or encryption routines and/or other data or programs of utility to many users. For example, in consumer applications, the read-only portion might include catalogs, advertising, or other commercial information. The read-write portion or write-once read-many portion would typically contain the unique user access key codes and unique user encryption keys (if used) and any other information unique to the particular user.

In a CD-ROM implementation, the read-only portion of the users' disks could be imprinted quickly and economically by pressing. The individualized portion,

typically a write-once, read-many portion, would then be quickly recorded on an appropriate recording CD-ROM drive. This approach may prove advantageous in a variety of high-volume applications.

If the user's portable storage medium key according to the present invention is re-writable, the medium may be "recharged" with new keys. Examples of such media keys are semiconductor memory units or cards, rewritable CD-ROM disks, floppy disks, and the like. In one implementation, a user key comprising a portable storage medium with less capacity can be "recharged" from another user key comprising a portable storage medium of greater capacity. For example, a user's memory card key could be re-charged from that user's CD-ROM key. Alternatively, a portable storage medium key can be re-charged at a secure computer, workstation, terminal, or facility. A yet further alternative is an exchange program wherein a user's used-up portable storage medium is exchanged for a re-charged or fresh storage medium with a new supply of keys. Other methods will be readily apparent to those skilled in the art.

Conventional authentication means or any of the authentication means of the invention can be used to insure that only the proper user with the proper storage key can re-charge same.

Additionally, if the portable storage medium key of the invention is also used as a credit or debit disk or unit or card or the like, it may be re-charged with additional funds or the like. In addition, transaction information could be logged onto the portable storage medium, either as verification, or for later down-loading; e.g. if the card or portable storage medium is used with systems that do not contact the server of the secure system of the invention; e.g. systems that are not connected to a network.

The present invention may also be incorporated in a portable electronic device. The portable electronic device may comprise portable storage media for storing the ultra-long identification keys and/or a database or databases of identification keys or pads of same, and/or the user s encryption key or keys or database or databases of encryption keys, or pads of same. A microprocessor and/or

logic circuitry, hereinafter referred to as a microprocessor, may be incorporated in the portable electronic device. For many forms of memory IC, a simple microprocessor can be fabricated on the same IC at small or negligible cost. If the portable storage medium is a portable hard disk drive, the microprocessor or logic functions can typically be implemented by adding additional code or programming to the microprocessor already present in the hard disk drive; again the cost would be negligible.

The microprocessor can provide additional security functions: additionally, it can optionally implement any of the functions that we have discussed as being implemented by a user terminal program or user access program running on the user's terminal or PC, and/or as contained on the user s portable storage medium. These include but are not limited to authentication, protocol, encryption, and other security functions. Advantages include off-loading these functions from the user's PC and thereby improving speed, simplifying the software, and providing additional assurance that these functions will be performed and not defeated, e.g.

by a rogue program or virus on the user's PC. For example, the microprocessor may be used to keep track of which keys have been used, typically by writing to the portable storage medium, as hereinafter described.

Conversely, any of the tasks described here as being performed by the microprocessor and/or clock of a portable electronic device may be performed by the microprocessor and/or clock of the remote terminal.

Additionally, the microprocessor can be programmed to "re-charge" the storage medium with new keys, per above, including the relevant security precautions. Additionally, the microprocessor can be programmed to log transaction information, per above, e.g. for stand-alone use in situations where the user uses the portable electronic device to conduct transactions without access to a PC. In many implementations of such portable electronic devices, the microprocessor will provide additional security to prevent unauthorized individuals or software from accessing or copying or using the identification keys on the portable storage medium.

For example, the microprocessor may be programmed to request a password from the user whenever the user attempts to access the identification keys on the portable storage medium. In order to access or use the identification keys on the portable storage media, the user must enter his or her appropriate user identification password. The user may enter the password through the remote terminal or a keypad on the portable storage media. In a preferred embodiment, the user s password is stored on the portable storage medium, in which case, the microprocessor is typically programmed to compare the user s stored password with the password entered by the user.

The microprocessor may refuse access to the identification keys on the portable storage medium for a fixed period of time if an incorrect user password is typed in or if several incorrect user passwords are typed in consecutively. For example, the microprocessor may prevent access to the identification keys on the portable

storage media for an hour when three incorrect user identification passwords are typed in consecutively.

The portable electronic device may further comprise a clock. The clock could be used, for example, to time the duration for which the microprocessor refuses access to the portable storage medium, as described hereinabove. Alternatively, the duration of refusal could be timed by a software timing loop or by keeping a running sum of the (known) execution times of each of the functions executed by the microprocessor. Clock circuits are inexpensive, and use little power. They can readily be powered for years by a small watch battery or the like. If the portable storage medium is a semiconductor memory, a clock circuit can readily be incorporated on the same IC as the memory and microprocessor.

Additionally or alternatively, the microprocessor may disable access to the portable storage medium if multiple incorrect user passwords are typed in consecutively. For example, the microprocessor may disable access when ten incorrect user passwords are typed in consecutively. Re-

enabling the system might require human intervention from the central server or provision of a special password or erasure of the contents of the portable storage medium.

Additionally or alternatively, the microprocessor may limit the number of passwords or one-time-pad entries accessed from the portable storage device in any pre-specified amount of time. This would prevent rapid copying of the identification keys stored on the portable storage medium. Again, a timing function. such as a clock or a software timing loop or running sum of execution times is required.

For example, the microprocessor may be programmed to limit the number of identification keys or one-time-pad entries or the like accessed from the portable storage device in a given number of seconds, minutes. hours or days. Alternatively, the microprocessor could prevent accessing identification keys or one-time-pad entries at a rate faster than they would be used by the user's terminal program. or could prevent accessing one-time-pad keys at a rate faster than the maximum transmission rate between the remote

terminal and the host computer. Other desirable rate limitations will readily be apparent to those skilled in the art.

Alternatively or additionally, the microprocessor may be programmed to output a time-dependent identification key or one-time-pad entry or the like; i.e. it may output a number that depends upon the time-of-day (including date) from the clock as well as upon the contents of the portable storage device. For example, the memory location or key accessed might depend upon the time-of-day, rather than the memory locations or keys being accessed in sequential order; i.e. the microprocessor selects the appropriate number or key from the portable storage medium based upon the current time. If desired, the microprocessor may combine the entry accessed from the portable storage medium with time-of-day information. For example, the keys could be added or concatenated and the result encrypted by the microprocessor and sent from the portable electronic device. Other time-dependent key techniques will readily be apparent to those skilled in the art. For any

time-dependent key technique, the host computer (server) would correspondingly be programmed to expect the result to be used as an identification key or as an encryption means; accordingly, the number outputted by the portable electronic device would be usable only at the time it was obtained.

The time-dependent key techniques can be used with any portable storage medium to produce keys that are only valid when produced; if the portable storage medium does not have its own local microprocessor. the above algorithms or similar algorithms can be implemented on the processor in the user's terminal or PC. They can also be used on the host computer or server; with the above algorithms or similar algorithms being implemented by the server or an outboard microprocessor, possibly associated with the key storage means. Accordingly, the above time-dependent key techniques can be operated in reverse to authenticate the server to the user in addition to authenticating the user to the server.

Additionally or alternatively, the microprocessor may be programmed to access the portable storage medium only if the user terminal program is running on the user's machine. For example, the microprocessor can require an access protocol or password (possibly incorporating time-of-day information).

Additionally or alternatively, the microprocessor may be programmed to access the portable storage medium only if the user terminal is accessing the host computer. For example, it might require an encrypted time-of-day function from the host computer. In addition to limiting access to the portable storage medium to legitimate requests, this authentication function would typically be made available to the user's terminal program.

Additionally, the microprocessor may optionally be programmed to prevent unauthorized accesses to the portable storage medium while it is being shipped to the user or when it is otherwise not in the immediate possession of the user. One way is to program a time lock function wherein the microprocessor will not grant access for a given

time interval or until a specified date-and-time. If the unit is received before that time, the user knows that there have been no accesses and his keys are safe.

Another way is to require a password or authorization sequence entered either by the user or the user terminal program or by a central server via a connection to the user s computer. For example, the card might be unlocked via an Internet connection to a server. Additionally, any of the other methods hereinbefore described may be used.

Note again that the unit may also optionally be programmed to erase its contents in response to one or several illicit attempts to obtain access or in response to an attempt to physically open the unit.

Further, the microprocessor can readily be programmed to log all accesses, typically on the portable storage medium or on the server or both.

Note that our portable electronic devices comprising the above time-lock, authorization-lock or authentication-lock, or access logging techniques can also be used for

secure delivery of one-time-pads or encryption keys, or private data or information, or the like. The other security means contemplated for our portable electronic devices may also be used in this type of application. Note additionally, one may provide additional security by spreading the information between two portable electronic devices or portable storage media in such a way that the contents of the two must be combined to give the information; for example one may encrypt the contents of one using a one-time pad stored in the other. The two are then shipped by different means or at different times (e.g. the second may optionally be shipped only after the first has been safely received).

Note also that our portable electronic devices may be implemented in a variety of form factors, including but not limited to any black box form factor or computer peripheral or computer plugin form factor (including a PCMCIA form factor or computer card form factor) or a form factor suitable to an ID card or badge or an access card or transaction card or credit card or the like.

There are many ways to connect peripheral devices or electronic storage media to a terminal or computer. Accordingly, an electronic portable storage medium or a portable electronic device, in accordance with the principles of the invention, may further comprise a PCMCIA interface or a serial port or a parallel port or SCSI port or USB (Universal Serial Bus) port or "Firewire" port or infrared link or radio link or a "memory reader" or any other port or communication means capable of enabling it to pass information to and receive information from the user's terminal or computer. Preferably, for a portable electronic device or memory medium which communicates with the remote terminal via an infrared link or radio link, the transmissions between the portable electronic device and the remote terminal are encrypted. The processor and/or logic circuitry in the portable electronic device may also optionally handle communication with the user's computer or PC.

In another embodiment, the portable electronic device may additionally record which identification keys and/or one-time-pad entries or the like have been previously

accessed, hence presumably used; rather than or in addition to the user's terminal program or access software performing this function. If the portable storage medium is rewritable, the identification keys, one-time-pads, and the like, may be overwritten once used. Alternatively, if the portable storage medium is writable, a usage sequence number or a usage record, table or list may be kept on the portable storage medium. Or a flag bit or field associated with the keys that have been used may be set or overwritten. Alternatively, e.g. if the portable storage medium is read-only, the portable storage device may further comprise a secondary writable portable storage medium, and a usage record, table or list may be kept there. This prevents an identification key or one-time-pad entry or the like from being used more than once.

Usage records can be alternatively kept in the server or host computer. Preferably usage records can be kept in both places and any discrepancy between usage records on the user's portable storage medium and on the server would suggest an attempt by a third party to illicitly gain access.

Again, such a discrepancy might be indicated by any attempt by either the user program or the server to re-use a one-time-pad key or one-time-pad entry that has already been used with the server or user program respectively. Such a discrepancy may also be indicated by any attempt to use a key or pad entry out-of-sequence or any other "out-of-synch episode".

Anything that suggests an attempt to gain illicit access either to the contents of the storage medium or anything that suggests an attempt to illicitly gain authorization with the server or anything that suggests a "man in the middle attack" by either counterfeiting the server to the user or the user to the server might be detected either by the portable electronic device or user program or by the server. Such suggestive incidents include those discussed above: e.g. repeated incorrect passwords typed in by the user, an attempt to access too many keys from the portable storage medium, an attempt to use a time-sensitive key at a later time, any failure to authenticate the user to the server or vice-versa, an attempt by either the alleged user or alleged

server to re-use a one-time password or any other "out-of-synch" episode, and the like.

Additionally, any attempt to use a known stolen user key or invalid or counterfeit user key suggests an attack, as does a usage pattern that suggests a user key may be stolen.

In one implementation, if the incident is detected by the portable electronic device or the user program, either or both may be programmed to contact or otherwise to notify the server. If the incident is detected by the server, the server may be programmed to contact the portable electronic device or user program.

Procedures for dealing with a suspected attack include but are not limited to: blocking access to the portable memory medium or portable electronic device for a set time; disabling access to the portable memory unit or portable electronic device or erasing the portable memory unit or portable electronic device (especially if the incident suggests that the unit or device has been stolen); blocking access by that device or unit or user to the server, either for a set time or until the situation is resolved (e.g. by the

server operator); notifying the true user (e.g. by E-mail or telephone to the true user); or notifying the server operator. These procedures may typically be implemented by appropriate programming for the portable electronic device s microprocessor and/or in the user program and/or in the server(s).

In another embodiment, the portable electronic device comprises a portable storage media, a microprocessor, and a modem. The microprocessor may handle the authentication protocols with the server. Additionally, the microprocessor may handle all encryption of information transmitted by the remote terminal via the modem and all decryption of information received from the host computer by the remote terminal via the modem. Including a modem also allows the microprocessor to be programmed to allow the user to conduct stand-alone transactions via the network and without later downloading when the user does not have access to a regular terminal or PC.

There are a variety of additional techniques and embodiments of the present invention that can be implemented using a CD-ROM key or any portable storage medium key or a portable electronic device key, per above.

For example, in another embodiment of the present invention, the host computer (server) may request an identification key from a random location on the user's CD-ROM or portable storage medium. The remote terminal or portable electronic device would read the identification key from the appropriate location in the memory medium and it would be transmitted to the host computer.

The present security system, in a most preferred embodiment, entails the use of a "double-sided" key technique comprising the use of separate entrance and exit keys at the beginning and end respectively of the communication session. In this method, the remote terminal program (or portable electronic device) transmits an identification key to the host computer (server) at the beginning of the communication session, thereby authenticating itself to gain access. The remote terminal

(or portable electronic device) transmits a second identification key to the server at the end of the communication session; typically, this can be used to validate the session. For example, the server is programmed such that it will not process the information transmitted unless both identification keys are correct; e.g. in a transaction system, the user transactions would be received by the server during the session, but not accepted or validated or processed unless or until a valid exit key is received at the end of the session. Thus, the first key functions to grant provisional access; the second key functions to provide the final authorization for the transactions.

In order to authenticate the server to the user's terminal program (or portable electronic device), the server would transmit to the user separate entrance and exit keys in a directly analogous manner; one would typically then have an exchange of identification keys between the user program and the server at both the beginning and the end of the communication session. Additionally, the

identification keys may be time-dependent, e.g. using the techniques described hereinabove.

Authenticating the user and server to each other at the beginning and end of the session blocks attempts to simply "hijack" the communication session. However, it does not block attempts to insert information into or delete information from otherwise valid sessions. The use of time-dependent identification keys imposes the further constraint that any tampering must be done in real time, and also blocks attempts to obtain valid entrance and exit keys, e.g. by using "man in the middle" techniques, and use them later. Thus authentication means to authenticate the entire session, or at least authenticate critical portions of the session, such as transaction information or transaction requests should always be included for maximizing security.

To authenticate a session or the critical portions thereof, the session or critical portions thereof may be encrypted. An unbroken encryption technique will serve to authenticate the encrypted messages or information. Thus, for example, encrypting the critical portions of the session

using a one-time-pad stored on the portable storage medium key and in the central server will authenticate that information.

Another technique to authenticate a session or the critical portions thereof is to calculate one or more check-sums or check-functions (hereinafter called check-functions) using means whereby it is (a) difficult or impossible to counterfeit the check-functions, and (b) difficult or impossible to fabricate spurious messages with the same check-function(s) as intercepted legitimate messages.

With the double-sided key technique, such check-functions can be included in or combined with the above-discussed end-of-session key. The end-of-session key will not only authenticate the user or server but it will also authenticate the contents of the session. For example, if the check-function(s) from the user to the server include the transaction information from the user, it authenticates that transaction information; if it includes the relevant messages from the server as well, it authenticates those as well: thus confirming that the user received the messages sent by the

server. Moreover, if the check-function includes the entire session, it authenticates the entire session. If the check-function(s) include time-of-day information, either for the communication session or for individual messages, it authenticates that time-of-day information as well.

It is preferable to combine the check-function(s) with the end-of-session key using a combining function that has good "diffusion", so that an attacker cannot separate the check-functions from the end-of-session key and attack them separately. Note, for example, that simply adding (without carries or with) the check-function to the end-of-session key cannot be reversed by an attacker if the latter is from a one-time key-pad. Other techniques include convolution or encrypting the combination of the two, using an encryption algorithm with good diffusion.

It is preferable that the combining function(s) have good "diffusion" so that it is not possible for an attacker to discover that some bits involve only the password, some bits involve only the checksum(s) and some bits only involve the time-of-day information; a combining function with good

diffusion helps scramble them all together. Good diffusion may be achieved by simply adding the check-sums or functions to part or all of the exit signature. or convolve them or use other algorithms that mix the information. Yet another means would be to encrypt the two together with an encryption function that has good diffusion.

There are many different ways to calculate check-functions that are difficult to counterfeit and where spurious messages with given check-functions are also difficult to counterfeit, or equivalently, spurious messages with the same check-functions as an intercepted message are difficult to counterfeit. For example, one may assign different parts or characters or pieces of the transmission different weights: e.g. depending on a key or random number from our portable storage medium or from the server.

There are various ways to combine a weight function with the messages or portions thereof; one way is to binary add without carrying on a bit-by-bit basis; another way is to group and multiply or group and add. typically throwing away the higher-order bits. One then typically sums the

results of these operations. The checksum or check function typically would either be that sum, or, preferably, the lower-order digits or bits of that sum.

In addition or alternatively, one may use changing, unique, or one-time weight functions. For example, one may have a region of the user's one-time-pad (or a separate pad) set aside for use as a weight function, and vary the starting point, or the order in which the entries are taken, or both from session to session. One way of doing so is to have the starting point or order or both depend upon a number taken from a one-time-pad or provided by the server or user program or calculated from time-of-day information, etc. Since there are N! orderings for a set, the same pad can (optionally) be re-used at little risk (especially if only the lower-order bits of the sum are used, per above). Yet another way to calculate a checkfunction is to encrypt the message(s) locally, then calculate the checkfunction on the encrypted messages.

Alternatively or in addition, the checksum or check-function may be combined with or include time-of-day

information or (better) a function computed from time-of-day information.

The double-sided key or password technique of the invention can use keys or passwords from our portable storage medium, or from an algorithm or from a string sent out by the server, or by any other means of generating passwords. For example, the double-sided keys could be implemented by a unique algorithm for each user: for example, by encrypting the time-of-day with a key unique to each user.

The second password can come from the same database or circuit or algorithm as the first, or from a separate database or circuit or algorithm than the first. Note also that we can use two or more passwords, either from a single portable storage medium, database, circuit or algorithm, or from two or more algorithms, or two or more circuits or ICs.

Alternatively, in lieu of a second key or password, the key or password is divided; one part is sent at the beginning

of the session and the second part is sent at the end of the session.

The double-sided key technique conserves keys and therefore is particularly suitable for implementations of the new security system invention using semiconductor memory keys or portable electronic device keys with semiconductor memories, and for other implementations of the invention using portable memory media of limited capacity.

We further contemplate user authentication and/or encryption means comprising more than one portable electronic device and/or portable storage medium, or, when possible, optionally using a single portable electronic device or portable storage medium to achieve the same ends . As described above, one implementation of our portable means to provide separate entrance and exit keys is to have two separate circuits or portable storage media or portable electronic devices; one for the entrance key, a second for the exit key, with both packaged together, e.g. in the same card-like configuration. Similarly, in an application like an electronic bank card, we may optionally have two or more

different chips, circuits, databases or storage media, algorithms, or programs; e.g. one that can be accessed by any merchant, and a second that can only be accessed by the issuing bank. Analogous applications for this type of device in other areas of commerce will be readily apparent. There may also optionally be two or more different connection means or protocols or keys or passwords or algorithms or authentication means, to use or access either or both of the databases or portable storage media, algorithms, or programs, chips or circuits. Similarly, the microprocessor in the portable electronic device may be programmed to provide different access privileges to different entities or programs or individuals accessing the device; e.g, in a bank card application, only the bank's program would be allowed to add funds to the total(s) stored in the card; e.g. doing so would require the bank's password or passwords for accessing a card or for accessing that individual's card, while any merchant to whom the user gave his card would be able to deduct. (For portable storage media keys or encryption means, the same function

would be implemented by an appropriate programming in the user terminal program.) Additionally, we previously described means for providing different access privileges or use privileges to a central server for different users; we further contemplate using such means to provide different access or use privileges to a user s portable electronic device or portable storage medium for different entities or programs or different authorized individuals. Note that this includes providing access to different services or functions, both in the case of the central server and in the case of a portable electronic device or portable storage medium.

As a special case, note that storing different passwords in the portable electronic device or portable storage medium allows us to provide programming to allow different users to access different authentication means or keys, different encryption means or keys, different data, and/or different functions on the same portable electronic device or portable storage medium.

Additionally, we contemplate the use of two or more portable storage media or portable electronic devices in the

same unit to allow said unit to be accessed by more than one means or to implement different functions. For example, it is now possible to buy CD-ROM business cards; these are read-only CD-ROM disks that have been trimmed to the size and shape of a business card.

Unfortunately, the prior art does not include writable CD-ROMS of this type; when one attempts to trim a writable CD-ROM, the two pieces of plastic separate: the writable film acts as a parting agent. We have found that it is possible to produce writable CD-ROM cards by providing clamping means to hold the top and bottom of the disk together while cutting off the rest of the disk to leave a card-shaped writable CD-ROM. (Note that it is expedient to have one clamping surface at least slightly resilient to provide an even clamping pressure.) We can then glue or fuse the exposed edges together, preferably before releasing said clamping means. (When fusing, it is often expedient to fuse beyond the writable layer; one way to do this is to use a slitting saw to remove a little bit of the writable layer, for example 10 or 20 mils around the outside. Another way is

to, when fusing, deform the plastic material outward, with heat and/or pressure, so as to fuse beyond the outside edge of the writable layer. Note that if the writable layer should break up and enter the fused region, one still obtains fusion between islands of the writable layer.) Said writable CD-ROM cards can be used as our portable storage media, as previously described for writable CD-ROMs. We further contemplate optionally embedding a portable electronic device, preferably in IC form, in the unused area of the card or attaching it to the surface of said card, preferably on the top or unused surface. The resulting device can be read either by a CD-ROM drive or by electronic means. We further contemplate optionally including magnetic storage media, typically on the unused ends of the card; the resultant device can then also be read by magnetic means. We further contemplate optionally providing the equivalent of a one or two-dimensional bar-code pattern, typically again on the unused ends of the card. Said pattern may either be printed or may be implemented using the writable CD-ROM material; the latter may optionally be done in a

recording CD-ROM drive. Note further that the two or more different storage means can store the same or different information to implement the same or different functions.

In addition, any of the techniques to detect and foil an illicit attack, including attempts to break into the system or steal identification keys or hijack a communication session may be combined with a program to create an entrapment session to keep the attacker or intruder linked or on the line to allow his call to be traced with a view to identifying or apprehending him. The entrapment session might also include programming to elicit additional information from the attacker or intruder or his computer or from servers along the way.

Additionally, any of the portable storage means of the present invention, including portable electronic devices. might also comprise programming to plant a "cookie" or information packet or a covert program or "identification virus" on an attacker's computer to facilitate subsequent identification of the attacker or at least of the computer he

or she used. The cookie or covert program would be planted upon detection of an illicit attack by any of the means described above. Such programming may be a part of any user terminal program or user access program on the portable storage means or needed to run the portable storage means; it may also be a part of any software drivers needed to access same, or of any other user program included on or with our system. (The only function of the "identification virus" is detection and identification of attackers; it is a totally benign virus. It should be programmed to be non-executing unless queried or activated, e.g. by a server.)

Alternatively, upon detection of an illicit attack, the server will plant the cookie or covert program in the attacker's computer, using means well-known to those skilled in the art (conventional cookies are normally planted by central servers and not by programming on the user's computer).

Alternatively, a service provider's computer may be programmed to plant such a cookie or covert program upon

MISSING AT THE TIME OF PUBLICATION

MISSING AT THE TIME OF PUBLICATION

attacker's session, plus other sessions immediately after (e.g. in case the attacker uses the legitimate user's machine for something else; e.g. to send an E-mail.)

A particular advantage of the "identification virus" approach is that it is typically attached to an existing program and is not detectable as a separate file. Similarly, the tracer cookie of the invention might be appended to an existing cookie. Alternatively, other means to hide the covert program or cookie or the like may be used; these include but are not limited to creating one or more hidden files, masquerading as a system or application file, marking it's block on the disk as "unusable" (and reversing same when one attempts to read it) and the like. A virus might be non-executing unless queried or except under other restricted circumstances.

Additionally, such "cookies" or markers or programs could be planted in any of the intermediate servers or machines along the way; for example, this would allow the maintainers or operators of the servers to notify an Internet service provider that the service is being attacked.

We further contemplate encrypted networks or adding encrypted network functionality to existing networks or to conventional networks. Thus. authentication and encryption means such as those discussed hereinbefore can be implemented between servers or routers or switches as well as between users and servers. They may also be implemented between private or corporate WANs or other relatively secure networks, when messages must be passed over an insecure network, such as the Internet. For example, a message may be unencrypted (or lightly encrypted) between the WAN users and their WAN servers, with the servers handling the encryption/decryption function for communication over the Internet between the servers.

We also contemplate encrypted networks (or encrypted network functionality) wherein the network servers or routers or switches comprise programming and/or auxiliary processors and/or hardware means to translate between one user's encryption and another's. This encryption translation function can also be implemented on a central server connected to the network, or on any

computer connected to the network and accessible to the users who wish to communicate, or on an auxiliary processor or hardware connected to said server or computer, or in our portable electronic devices or in stationary versions of same or plugin versions of same. This encryption translation technique enables any two users to communicate without... knowing each other's encryption means or keys; this avoids the need for each user to know every other user's encryption means or keys, which quickly becomes both cumbersome and, for most cryptosystems, insecure as well.

For example, we contemplate one-time-pad encrypted networks wherein the central servers or routers or switches comprise means to translate between any user's one-time-pad and any other user's one-time-pad. One way to do this is for the server to use the first user's one-time-pad to decrypt the message, and then use the second user's one-time-pad to re-encrypt the message. A second way to do this is to first calculate a "translation pad" from the two users' one-time-pads and then use the translation pad to translate the message from one user's

one-time-pad to the other user's one-time-pad. The second approach has two advantages. First, the message never exists in decrypted form in the central server or router or switch. Second, the translation pad can be calculated by an utterly secure "pad server", then supplied to the network server or router or switch to translate the message between the two pads. This means that the message is never in a network server that also has the information required to decrypt it.

Similarly, for other encryption means, the translation processor may either (a) decrypt one user's message, then re-encrypt it using the other user's encryption means, or, alternatively (b) generate a translation function or algorithm between the two users' encryptions. The second approach again has the advantage that the message never exists in decrypted form while being translated. Similarly, the translation algorithm, function, or key or the translation function itself may be calculated or performed in a secure translation server connected to the network server.

In addition, it is desirable to avoid the need for each network server or router or switch to know every user's encryption means or keys. Therefore, we contemplate networks wherein the network servers or routers or switches comprise means to translate between each of their user's encryption means and "trunk" encryption means for communicating with other servers. Thus, when a first user served by a first network server communicates with a second user served by a second network server, the first network server translates from the first user's encryption means or key to an intermediate encryption means or a "trunk" encryption means or key which the second network server translates to the second user's encryption means or key.

Implementing this with one-time-pad systems entails translation means directly analogous to those described for translating between two users directly; the main difference is an extra translation step.

In addition, we contemplate encrypted networks wherein the network servers or routers or switches comprise means to send the same encryption means or keys to any

two users who wish to communicate. The encryption means or key or keys may be sent as part of the call setup or link initiation, or it may sometimes optionally be sent ahead-of-time in anticipation of being needed. Typically, the encryption means or key would itself be encrypted, e.g. using each user's encryption means. If the two users are on secure networks (e.g. a corporate WAN) connected via an insecure network such as the Internet, the encryption means would be encrypted when being passed over the insecure network, but might even optionally be unencrypted when being passed over the secure networks to the users. Note, however, that the secure network may include its own encryption, in which case one may optionally translate to that network s encryption.

More often, each user program or portable storage medium or portable electronic device will comprise encryption means or keys known to its network server or central server, which will use said user's encryption means or keys to encrypt the means or keys for communicating with the other user. Again, the information would most often

be passed as part of the call setup or link initiation process; if further keys are needed, they might be transmitted during the session.

For example, in a one-time-pad system, each user's server may use that user's one-time-pad to encrypt a new, shorter one-time-pad that is sent to both users and used to communicate between the two. As each one-time-pad entry can be used to communicate a single one-time-pad entry, one is effectively turning a portion of each user's one-time-pad into a small one-time-pad common to the two users for communication between the two users. Alternatively, the one-time-pad entries can be conserved by using each of them to transmit a single key to be used in one of the conventional mathematical encryption means for a subset of the session or for the duration of the session.

Further, for small numbers of users, or for demonstration purposes, it is feasible to provide each user with a custom CD-ROM or other portable storage means comprising different tracks or storage areas, each comprising encryption means or keys for one of the other users. For

example, in a one-time-pad system, each user may be issued a CD-ROM disk with tracks or storage areas comprising different one-time-subpads for each of the other users. Preferably, in all cases user "A's track or pad for user "B" is the same as user "B's:" track or pad for user "A" and is different from all other tracks or pad on all of the disks.

Note also that this technique can be extended to a moderate number of users, e.g. if the one-time-pad entries are conserved by using each of them to transmit a single key to be used in one of the conventional mathematical encryption means for a subset of a communication session or for the duration of a communication session.

Finally, we also contemplate translating authentications and authentication messages between users as well, typically by means analogous to the means described above for translating messages between different users' encryption systems. For example, if each user has a one-time-pad of authentication keys, the central translation program or means might translate each user's authentication into the next entry on the other user's authentication pad.

Similarly, any of our means for authenticating a user to a central server can be used to authenticate two or more users connected to said central server, e.g. when the users are authenticated to the central server, the central server sends an authentication message for each user to the other user's terminal programs. Similarly, when we use the central server or other processor or hardware to translate messages between individual users' encryption means, we also optionally contemplate appending or including an authentication message or the translated authentications. Finally, we contemplated authentication means where the user programs authenticate themselves to the central sever by using a key or encryption means unique to the user to encrypt the user's claimed identity or other identification message along with a variable padding, such as a random padding or a predictably varying padding, such as the date and time. One way of translating these authentication means between users is to simply translate the encryption means, e.g. by the means described previously.

It is also desirable to use conventional encryption means or one-time-pad encryption means or other encryption means to transmit pictures. However, a picture can use up one-time-pad entries quickly. There are several ways to deal with this. One way is to conserve the one-time-pad entries, again, by using each of them to transmit a single key to be used in one of the conventional mathematical encryption means for a subset of a picture or for one or more pictures. Another way is to compress the picture before encrypting it, e.g. with a one-time-pad: digital compression tends also to remove redundant information, thereby making unauthorized decryption more difficult.

A second class of techniques is the digital analog of shredding the picture or painting over it with random splatters of paint. One "shredding" technique is to divide the picture into small squares or hexagons or other small groups of pixels and use a series of random or quasirandom numbers to scramble their positions and optionally to randomize their orientations. For example, if a picture is divided into 100 by 100 small squares, there are 10,000

squares in all; approximately 13 binary bits are needed to assign each square a random or quasi-random position within the 100 by 100 array, and another 2 bits may optionally be used to pick one of (4) random orientations. The total is 17 bits per square or 170,000 bits in all or 21,000 bytes of random or quasi-random numbers to digitally shred (and unshred) one picture.

Those random or quasi random numbers can come from one of our one-time-pads. Alternatively, one of our one-time-pad entries or some other encryption means can be used to transmit a "seed value" for a random-number-generator algorithm or other position-scrambling algorithm. If the one-time-pad is used, 21,000 bytes per picture means that one 650Mbyte CD-ROM can be used to shred and unshred 30,500 pictures.

A functionally similar positional scrambling technique is to append to each pixel or block of pixels a number or numbers describing it's location in the picture, then encrypt said location number, then re-order the pixels or blocks

according to their encrypted location numbers. Essentially, one is encrypting the locations rather than the signal itself.

One digital splatter painting technique is to divide the picture into small squares or hexagons or other small groups of pixels and use a series of random or quasirandom numbers to randomly add to or change each group's chrominance (color) and luminance (brightness). Note that one bit can optionally be reserved to randomly invert or not invert either the brightness or any of the colors. The bits-per-picture calculations are similar, and the random or quasi-random numbers can come from a one-time-pad or random-number- generator algorithm or other color scrambling algorithm in a directly analogous manner the digital shredding technique.

Note that one may also apply an invertible smoothing or smear function either to the digitally scrambled or digitally splattered picture on the one hand or to the scramble function or splatter function on the other. This will hamper any attempt at applying computerized edge-matching algorithms.

Analogous techniques may be applied to the compressed picture information; again, one has the choice of encrypting the picture information or scrambling the positional information and, for blocks, scrambling the orientation.

It is also desirable to encrypt voice or audio signals, e.g. telephone, radio and cellular telephone signals, again using our user-unique encryption means. Again, one may elect to encrypt the compressed signal to reduce the amount of encryption needed or to save one-time-pad entries. A digital cellular telephone with a user-unique one-time-pad or encryption key pad on a miniature hard-drive is within the scope of this invention. Note also that the one-time-pad can periodically be "recharged" with new numbers or keys using means we previously discussed.

Alternatively or in addition, one can scramble sub-blocks or samples in time; equivalently, one may attach to each sample or block a number denoting its position in the datastream, encrypt said number, then re-order the samples or blocks according to the encrypted numbers.

Alternatively, one can add a random or pseudo-random "noise" signal to mask the voice signal. Alternatively, one may apply these techniques to the compressed signal or to the signal in the frequency domain instead of the time-domain..

It is also desirable to encrypt video signals, e.g. television programs, motion pictures, teleconferencing, and the like, again using our user-unique encryption means. Again, one may elect to encrypt the compressed signal to reduce the amount of encryption needed or to save one-time-pad entries. Alternatively or additionally, one may scramble pixels or blocks of pixels in position or orientation, as discussed for pictures, or in time (e.g. between frames), as discussed for voice signals. Alternatively or additionally, one can add a random or pseudo-random "noise" signal to mask the picture, as previously discussed. Again, one may alternatively apply these techniques to the compressed signal.

In addition, since many television programs and motion pictures are publicly available and not secrets, an

encryption or scrambling technique that only degrades the quality so that the program or motion picture is unwatchable may be sufficient. For example, a single one-time-pad entry can be used to seed a pseudo-random number generation algorithm used to generate numbers to partially or totally scramble or mask or encrypt the picture.

Finally, our user encryption keys or databases or pads of keys or one-time-pads and our associated systems and techniques can be used with any conventional means for encrypting, scrambling or masking digital picture information, or voice or audio or video information or signals.

The means discussed herein for securing and controlling access to a host computer or server or network or communication via a network can also be implemented on an auxiliary or dedicated processor or computer such as a "firewall processor", or on a network processor, router, or switching system, instead of the host computer or server. An auxiliary or dedicated processor or computer eliminates the need for the host computer to perform the

authentication, decreasing the processing load of the host computer.

The CD-ROM or other portable storage medium or portable electronic device can be used to control access to, through, or under the control of, any stored-program processor or computer capable of directly or indirectly accessing storage capacity sufficient to hold the requisite database of user key codes. Indirect access may comprise remote access via a network or may comprise access from another processor or memory system.

Note also that the novel techniques herein described can be used in applications where the keypad or key data or portions thereof are installed or copied onto the hard disk drive of the user's computer or terminal or otherwise stored or installed on the user's computer or terminal. Our techniques can also be used in applications wherein the portable electronic device comprises programming means to act as a user terminal.

The invention also encompasses a method to store secure versions of documents. In this system, documents or

files created by a depositor are transmitted within internal or external information network and are recorded in a digitally signed and encrypted format. Such data or files or documents can be encrypted by either OTP (one time pad) or other encryption means and recorded on preferably a write once read only storage device. Once the data is stored in such a "digitally frozen" state, it may never be changed or written over; thus it becomes part of a permanent record which can be stored in third party network or storage facility. The data is encoded and digitally locked via a digital signature or the like until confirmation of the data is needed. The third party storage can be located either at one or more remote locations or in a secured, "bonded" or controlled facility at a user location. The date and time of receipt of the data is preferably stored with the data. In addition, the storage site preferably follows physical security procedures to ensure an uncompromised chain of custody of the storage media. Once the storage of encrypted documents is completed, they

will enable the user to allow confirmation to any authorized viewer.

The keys necessary to access, decript and confirm the contents of the data can be held solely by the archive facility, solely by the depositor, or jointly by both the archive facility and the depositor. In this manner, access to the data can be controlled as desired. In addition, the keys necessary to access, decrypt and confirm the data can be delivered to another party.

In the case where both the depositor and the archive facility jointly hold the access keys, the archival system can require that the two sets of keys be submitted simultaneously, or within a predetermined trial period to grant access to the data.

The above document archival system can be implemented to record a series of communications between two users pertaining to, for example, a contract negotiation. In this embodiment, each communication in the series is stored by the archival facility in a digitally frozen form, e.g., via a digital signature. In addition, the signatures of some

or all of the communications can be verified by the archival facility or another verification source. This verification can be appended to a final communication which is also digitally frozen and stored at the archival facility.

Access to the data stored by the archival facility of any of the above methods can be limited by time of access and/or delivery and/or by location of the party requesting access or of the archival facility.

The portable storage media or portable electronic device can be programmed with biometric data such as fingerprint or eye retina scan or voice confirmation data to control access to and to confirm ownership. The device to which the portable storage media or the portable electronic device is attached can have programming and sensors to detect such biometric data. The Portable electronic device itself can include such sensors.

It will also readily be apparent to those skilled in the art that the means described herein for providing secure access to a host computer or server or to databases or transaction processing systems implemented on same can

also be used to control access to other computers, or to networks, or to databases or transaction processing systems or other programs or information functions implemented on or accessed through same. The read-write portion or write-once read-many portion would typically contain the unique user access key codes and unique user encryption keys (when used) and any other information unique to the particular user.

In a CD-ROM implementation, the read-only portion of the users' disks could be imprinted quickly and economically by pressing. The individualized portion, typically a write-once, read-many portion, would then be quickly recorded on an appropriate recording CD-ROM drive. This approach may prove advantageous in a variety of high-volume applications.

Although the foregoing description has been given by way of preferred embodiments, it will be understood by those skilled in the art that other forms of the invention falling within the ambit of the following claims is contemplated. Accordingly, reference should be made to

the following claims in determining the full scope of the invention.

We claim:

1. A method of providing user identification and authentication using ultra long identification key codes and/or ultra large databases of identification key codes in a manner providing secure access from a remote computer terminal to a database or server transaction program stored on a host computer, comprising the steps of:

(a) utilizing key generation algorithms and/or hardware to generate individual, class specific, or both user access key codes which may optionally contain individual encryption keys;

(b) utilizing key generation algorithms to generate one or more media access codes;

(c) creating a database or otherwise updating an existing database comprising a compilation of each of said individualized and class specific user access key codes which have been generated for predetermined authorized users of the server transaction program;

(d) recording, on separate individual portable storage media directly compatible with and readily insertable and removable from said remote computer terminal, each of said individualized and class specified user access key codes, along with the optional individual encryption keys, and the media access codes;

(e) loading or providing the server serving as the host computer with a complete registry or compilation of each individualized and class specified access key code and any optional individual encryption keys which have been generated by the key generation algorithms;

(f) providing each authorized user with said portable storage medium containing the authorized user's individual or class specified access key code or key codes;

(g) providing the server with computer programming including steps for comparing individual and class specified access key codes transmitted over telephone networks or communication networks from a user's remote computer terminal against the stored compilation of authorized access key codes and permitting correct matches

to have access to said server transaction program while denying access to unauthorized access key codes;

(h) providing users' remote computer terminals with programming including the steps for comparing media access codes entered by the user against media access codes stored on the portable storage medium and permitting correct matches to have access to the individual or class specified access key codes stored on the portable storage medium;

(i) providing users' remote computer terminals with programming to permit connection to said server through a communication network or telephone network and to transmit individual and class specific access key codes through said remote computer terminal utilizing readers for the portable storage medium to said server for the purposes of gaining access to said server transaction database; and

(j) conducting a communications session between the user's remote computer terminal and said server transaction program through said telephone or communication network.

2. A method of providing user identification and authentication using ultra long identification key codes and/or ultra large databases of identification key codes in a manner providing secure access from a remote computer terminal to a database or server transaction program stored on a host computer, comprising the steps of:

(a) utilizing key generation algorithms to generate individual, class specific, or both user key codes which may optionally contain individual encryption keys;

(b) creating a database or otherwise updating an existing database comprising a compilation of each of the access key codes which have been generated for predetermined authorized users of the server transaction program;

(c) recording, on separate individual portable storage media directly compatible with and readily insertable and removable from said remote computer terminal, each of said individualized and class specified user access key codes along with the optional individual encryption keys;

(d) loading or providing the server serving as the host computer with a complete registry or compilation of each individualized and class specified access key code and any optional individual encryption keys which have been generated by the key generation algorithms:

(e) providing each authorized user with said portable storage medium containing the authorized user's individual or class specified access key code:

(f) providing the server with computer programming including steps for comparing individual and class specified access key codes transmitted over telephone networks or communication networks from a user's remote computer terminal against the stored compilation of authorized access key codes and permitting correct matches to have access to said server transaction program while denying access to unauthorized access key codes:

(g) providing users' remote computer terminals with programming to permit connection to said server through a communication network or telephone network and to repeatedly or periodically transmit individual and class

specific access key codes through said remote computer terminal utilizing readers for the portable storage medium to said server for the purposes of gaining access to said server transaction database; and

(h) conducting a communications session between the user's remote computer terminal and said server transaction program through said telephone or communication network.

3. A method of providing user identification and authentication using ultra long identification key codes and/or ultra large databases of identification key codes in a manner providing secure access from a remote computer terminal to a database or server transaction program stored on a host computer, comprising the steps of:

(a) utilizing key generation algorithms to generate individual, class specific, or both user key codes which may optionally contain individual encryption keys;

(b) creating a database or otherwise updating an existing database comprising a compilation of each of the

access key codes which have been generated for predetermined authorized users of the server transaction program;

(c) recording, on separate individual portable storage media directly compatible with and readily insertable and removable from said remote computer terminal, each of said individualized and class specified user access key codes along with the optional individual encryption keys;

(d) loading or providing the server serving as the host computer with a complete registry or compilation of each individualized and class specified access key code, the location on the portable storage media where each individualized and class specified access key code is stored, and any optional individual encryption keys which have been generated by the key generation algorithms;

(e) providing each authorized user with said portable storage medium containing the authorized user's individual or class specified access key code;

(f) providing the server with computer programming including steps for

(i) requesting one or more individual and class specified access key codes from a specific location on the portable storage media, and

(ii) comparing individual and class specified access key codes transmitted over telephone networks or communication networks from a user's remote computer terminal against the stored compilation of authorized access key codes and permitting correct matches to have access to said server transaction program while denying access to unauthorized access key codes;

(g) providing users' remote computer terminals with programming to permit connection to said server through a communication network or telephone network and to transmit individual and class specific access key codes through said remote computer terminal utilizing readers for the portable storage medium to said server for the purposes of gaining access to said server transaction database;

(h) conducting a communications session between the user's remote computer terminal and said server

transaction program through said telephone or communication network.

4. A method of providing user identification and authentication using ultra long identification key codes and/or ultra large databases of identification key codes in a manner providing secure access from a remote computer terminal to a database or server transaction program stored on a host computer, comprising the steps of:

(a) utilizing key generation algorithms to generate individual, class specific, or both user key codes which may optionally contain individual encryption keys:

(b) creating a database or otherwise updating an existing database comprising a compilation of each of the access key codes which have been generated for predetermined authorized users of the server transaction program;

(c) recording, on separate individual portable storage media directly compatible with and readily insertable and removable from said remote computer terminal, each of

said individualized and class specified user access key codes along with the optional individual encryption keys;

(d) loading or providing the server serving as the host computer with a complete registry or compilation of each individualized and class specified access key code and any optional individual encryption keys which have been generated by the key generation algorithms;

(e) providing each authorized user with said portable storage medium containing the authorized user's individual or class specified access key code;

(f) providing the server with computer programming including steps for comparing individual and class specified access key codes transmitted over telephone networks or communication networks from a user's remote computer terminal against the stored compilation of authorized access key codes and permitting correct matches to have access to said server transaction program while denying access to unauthorized access key codes;

(g) providing users' remote computer terminals with programming to permit connection to said server

through a communication network or telephone network and to transmit individual and class specific access key codes through said remote computer terminal utilizing readers for the portable storage medium to said server for the purposes of gaining access to said server transaction database;

(h) providing users' remote computer terminals with programming to permit copying of information or one or more programs from the server or the portable storage medium to the remote computer terminals; and

(i) conducting a communications session between the user's remote computer terminal and said server transaction program through said telephone or communication network.

5. A method of providing user identification and authentication as described in claim 2, wherein in said step of providing users' remote computer with programming, individual and class specific key codes are transmitted at the beginning and end of the communications session.

6. A method of providing user identification and authentication as described in claim 2, further comprising the steps of:

(a) counting the information transmitted from the remote terminal to the server according to a pre-determined algorithm;

(b) transmitting the count to the server whenever individual and class specific access key codes are transmitted to the server.

7. A user identification authentication system using ultra long identification keys and/or ultra large databases of identification keys for secure remote computer terminal access to a host computer comprising:

(a) a host computer having a compiled database of pre-authorized user access key codes of ultra long length;

(b) a series of individual portable storage media directly compatible with and readily insertable and removable from said remote computer terminal, each containing

(i) a unique or class unique access key code distributed among authorized users of a server transaction program, and

(ii) one or more media access codes;

(d) a remote terminal with programing to compare entered media access codes with the media access codes stored on the portable storage media and to deny access to the access key codes stored on the portable storage media to any unauthorized media access codes but to permit access to any user entering an authorized media access code;

(e) a server with programming to compare received access key codes with stored authorized access key codes and to deny access to the server transaction program to any user transmitting an unauthorized key code but to permit access to any user transmitting an authorized access key code;

(f) each of said access key codes being ultra long and comprising at least 20 characters or digits (requiring 20 or 10 bytes, respectively).

8. A method to control access to a resource connected to a network accessible from a plurality of communication devices, comprising the steps of:

(a) generating a plurality of sets of unique or class specific key codes, said key codes being ultra-long and/or said sets of key codes being ultra-large;

(b) creating a complete key database or otherwise updating an existing complete key database comprising a compilation of each of said plurality of sets of unique or class specific key codes;

(c) providing a host device with said complete key database, said host device being connected to said network;

(d) recording, on one of said communication devices or on a unique individual portable storage media compatible with one of said communication devices, one of said sets of unique or class specific key codes;

(e) programming said one communication device to permit connection to said host device through said network and to transmit or receive individual or class specific key codes to or from said host;

(f) programming said host device to permit connection to one of said communication devices and to transmit or receive individual or class specific key codes to or from said one communication device;

(g) distributing said communication device or said unique individual portable storage media upon which said one set of unique or class specific key codes has been recorded to an authorized user of said resource.

9. The method of claim 8, wherein said keys are identification keys and further comprising programming said host device to compare individual or class specific identification key codes transmitted from said one communication device against the stored compilation of identification key codes, and to grant said one communication device access to said resource if said transmitted key code matches a key code in said complete database, while denying such access in the case of an incorrect match.I.The method of claim 2, wherein said complete identification key database provided to said host

device is stored in encrypted form.II.The method of claim 3, further comprising programming said host device to employ a trap door algorithm to compare an identification key transmitted by said one communication device with an encrypted identification key provided to said host device.

12. A method to control access to a resource accessible from a plurality of user devices, comprising the steps of:

generating a plurality of sets of one or more unique or class specific keys, said keys being ultra-long or said sets of keys being ultra-large;

creating a key database comprising a compilation of each of said plurality of sets of unique or class specific keys;

storing said key database on a storage medium connected to a host, said host including one or more programmable devices;

programming a user device to communicate with said host and to transmit or receive keys to or from said host;

programming said host to communicate with said user device and to transmit or receive keys to or from said user device; and

distributing one or more sets of user keys to an authorized user of said resource, each set of user keys comprising one of said sets of unique or class specific keys, and said key database including each of said sets of user keys; and

said set of user keys being recorded on a storage medium integral or compatible with said user device prior to or after distribution of said set of user keys to said authorized user.

13. The method of claim 12, wherein said set of user keys is recorded on said storage medium prior to distribution to said authorized user.

14. The method of claim 12, wherein said set of user keys is distributed electronically.

15. The method of claim 13 wherein said storage medium is a portable storage media.

16. The method of claim 12, further comprising:

said keys being identification or authentication keys; and

transmitting a key from said set of user keys to said host;

programming said host device to compare said transmitted key against said key database, and denying said user device access to said resource if said transmitted key does not match a key in said key database.

17. The method of claim 12, further comprising:

said keys being identification or authentication keys; and

transmitting a key from said set of user keys from said host to said user device;

programming said user device to compare said transmitted key against said set of user keys; and

terminating a communication session if said transmitted key does not match a key in said set of user keys.

18. The method of claim 16, wherein said set of user keys includes a plurality of groups of keys, each group of keys providing a different level of access to said resource.

19. The method of claim 18, wherein information associating said groups of keys with said different levels of access to said resource is recorded on said storage medium.

20. The method of claim 18, wherein information associating said groups of keys with said different levels of access to said resource is recorded on said host.

21. The method of claim 12, further comprising steps to prevent or detect attempted re-use of a key.

22. The method of claim 12, wherein said set of user keys is a one time pad, each key of said set of user keys being used only once; and further comprising steps to prevent or detect attempted reuse of a key.

23. The method of claim 12 further comprising:

said set of user keys is a one time pad, each key of said set of user keys being used only once;

programming said user device and said host to use keys of said set of user keys as seeds for a key generation algorithm to generate an encryption key.

24. The method of claim 12, further comprising:

programming said user device to encrypt a current time and date known to said user device using a key from said set of user keys and to transmit said encrypted time and date to said host; and

programming said host to decrypt said encrypted time and date using a key from said set of user keys and to deny access to said resource if said decrypted time and date is not within a predetermined tolerance of a current time and date known to said host.

25. The method of claim 12, further comprising:

programming said host device to encrypt a current time and date known to said host using a key from said set of user keys and to transmit said encrypted time and date to said user device; and

programming said user device to decrypt said encrypted time and date using a key from said set of user keys and to terminate a communication session if said decrypted time and date is not within a predetermined tolerance of a current time and date known to said user device.

26. The method of claim 21, wherein:

said prevention and detection steps comprise programming said host or said user device to track usage of said user keys.

27. The method of claim 26, further comprising:

programming said host to provide a pointer to said user device to a previously unused key; and

programming said user device to receive said pointer from said host, by which said user device can select said previously unused key from said set of user keys.

28. The method of claim 21 wherein said storage medium of said user device comprises writable memory, and said method further comprises recording usage of keys on said writable memory.

29. The method of claim 21, wherein said storage medium of said user device comprises writable memory, and said method further comprises erasing or overwriting used keys of said set of user keys.

30. The method of claim 21, wherein said steps to prevent or detect attempted reuse of a key comprise programming said user device and said host with an algorithm by which to select a previously unused key from said set of user keys.

31. The method of claim 30, wherein said algorithm comprises selecting a key from said set of user keys according to a current time and date.

32. The method of claim 31, wherein

said keys are identification or authentication keys; and

programming said host to deny access to said resource if a time and date associated with a key selected according to said algorithm by said user device is not within a predetermined length of time of a time and date of a key selected by said host according to said algorithm.

33. The method of claim 31, wherein

said keys are identification or authentication keys; and

programming said user device to terminate a communication session if a time and date associated with a key selected according to said algorithm by said user device is not within a predetermined length of time of a time and date of a key selected by said host according to said algorithm.

34. The method of claim 31, wherein:

said keys are identification or authentication keys;

programming said user device to transmit a current time and date known to said user device along with an associated key selected from said set of user keys according to said algorithm; and

programming said host to allow access to said resource if said time and date transmitted by said user device is within a predetermined amount of time of a current time and date known to said host, and if said associated key transmitted by said user device matches a key stored on said host, which stored key is associated with said time and date transmitted by said user device.

35. The method of claim 31, wherein:

said keys are identification or authentication keys;

programming said host to transmit a current time and date known to said host along with an associated key selected from said set of user keys according to said algorithm; and

programming said user device to terminate a communication session if said time and date transmitted by said host is not within a predetermined amount of time of a current time and date known to said user device, and if said associated key transmitted by said host matches a key stored on said storage media of said user device, which stored key is associated with said time and date transmitted by said host.

36. The method of claim 21, wherein:

said steps to prevent or detect attempted reuse of a key comprises programming said host to track usage of said user keys; and

further comprising programming said user device to request, in the event that said host rejects a key selected by said user device as being previously used, that said host transmit another allegedly previously used key.

37. The method of claim 21, wherein:

said steps to prevent or detect attempted reuse of a key comprises programming said user device to track usage of said user keys; and

further comprising programming said host to request, in the event that said user device rejects a key selected by said host as being previously used, that said user device transmit another allegedly previously used key.

38. The method of claim 30, wherein:

said steps to prevent or detect attempted reuse of a key comprises programming said host to track usage of said user keys; and

further comprising programming said user device to request, in the event that said host rejects a key selected by said user device as being previously used, that said host transmit another allegedly previously used key.

39. The method of claim 30, wherein:

said steps to prevent or detect attempted reuse of a key comprises programming said user device to track usage of said user keys; and

further comprising programming said host to request, in the event that said user device rejects a key selected by said host as being previously used, that said user device transmit another allegedly previously used key.

40. The method of claim 12, wherein:

said keys are authentication or identification keys;

programming said host to prevent reuse of a key of said set of user keys;

said user device selects a plurality of keys from said set of user keys at random;

said user device transmits a first key from said plurality of selected keys to said host; and

if said host rejects said first key, said user device transmits a second key from said plurality of selected keys to said host.

41. The method of claim 12, wherein additional or replacement keys are appended to or written over said set of user keys recorded on said storage medium after distribution of said set of user keys to said authorized user.

42. The method of claim 12, wherein said keys are generated by a random or pseudo-random algorithm.

43. The method of claim 12, wherein said keys are generated by a random physical process.

44. The method of claim 12, wherein said step of storing said key database on said storage medium comprises encrypting said keys.

45. The method of claim 16, wherein said transmitted key is compared to an encrypted key stored on said host by a trap door algorithm.

46. The method of claim 12, wherein a plurality of sets of user keys are distributed to said authorized user, and wherein said host identifies which one of said plurality of sets of user keys is to be used for a communication session.

47. The method of claim 12, wherein said user device must connect to said host to gain access to said set of user keys.

48. The method of claim 12, wherein said keys of said set of user keys are used according to a predetermined algorithm.

49. The method of claim 48, wherein said algorithm comprises selecting a key from said set of user keys according to a current time and date.

50. The method of claim 16, wherein an identification or authentication key transmitted to said host is encrypted prior to being transmitted.

51. The method of claim 50, wherein said transmitted key is encrypted with a selected encryption code selected from a database of encryption codes located on said user device and located on said host.

52. The method of claim 51, wherein said selected encryption code is selected by said host.

53. The method of claim 51, where said selected encryption code is selected by said user device according to a predetermined algorithm.

54. The method of claim 50, wherein said transmitted key is encrypted with a public key encryption code transmitted to said user device from said host.

55. The method of claim 50, wherein said transmitted key is encrypted with an encryption code compiled from a first partial encryption code transmitted to said user device from said host and a second partial encryption code stored on said user device.

56. The method of claim 50, wherein said transmitted key is encrypted with a transmitted encryption code transmitted from said host, said transmitted encryption code being padded by said user device with random, null or nonsense prefixes or suffixes prior to encrypting said transmitted key.

57. The method of claim 56, wherein said transmitted key code is padded according to an algorithm specified by said host.

58. The method of claim 50, wherein

said user device transmits a first identification or authentication key to said host, said first transmitted identification or authentication key being unencrypted or relatively lightly encrypted;

said user device transmits a second identification or authentication key to said host, said second transmitted identification or authentication key being relatively highly encrypted; and

said host uses said first identification or authentication key to determine an encryption code to use to decrypt and verify said second transmitted identification or authentication key.

59. The method of claim 17, wherein an identification or authentication key transmitted to said user device from said host is encrypted prior to being transmitted.

60. The method of claim 59, wherein said transmitted key is encrypted with a selected encryption code selected from a database of encryption codes located on said user device and located on said host.

61. The method of claim 60, wherein said selected encryption code is selected by said user device.

62. The method of claim 60, where said selected encryption code is selected by said host according to a predetermined algorithm.

63. The method of claim 59, wherein said transmitted key is encrypted with a public key encryption code transmitted to said host from said user device.

64. The method of claim 59, wherein said transmitted key is encrypted with an encryption code compiled from a first

partial encryption code transmitted to said host from said user device and a second partial encryption code stored on said host.

65. The method of claim 59, wherein said transmitted key is encrypted with a transmitted encryption code transmitted from said user device, said transmitted encryption code being padded by said host with random, null or nonsense prefixes or suffixes prior to encrypting said transmitted key.

66. The method of claim 65, wherein said transmitted key code is padded according to an algorithm specified by said user device.

67. The method of claim 59, wherein

said host transmits a first identification or authentication key to said user device, said first transmitted identification or authentication key being unencrypted or relatively lightly encrypted;

said host transmits a second identification or authentication key to said user device, said second transmitted identification or authentication key being relatively highly encrypted; and

said user device uses said first identification or authentication key to determine an encryption code to use to decrypt and verify said second transmitted identification or authentication key.

68. The method of claim 12, further comprising:

said keys including identification or authentication keys;

a first device comprising one of said host or said user device;

a second device comprising one of said user device or said host;

programming said first device to transmit an initial portion of a first identification or authentication key from said set of user keys;

programming said second device to compare said initial portion of said first identification or authentication key transmitted by said first device with an initial portion of a comparison key recorded on a storage medium of said second device; and

programming said second device to transmit, in the event of a correct comparison, an initial portion of a second identification or authentication key from said set of user keys to said first device.

69. The method of claim 68, further comprising:

programming said first device to compare said initial portion of said second identification or authentication key transmitted by said second device with an initial portion of a comparison key recorded on a storage medium of said first device; and

programming said first device to transmit, in the event of a correct comparison, an further portion of said first identification or authentication key to said second device.

70. The method of claim 69, further comprising:

programming said second device to compare said further portion of said first identification or authentication key transmitted by said first device with a further portion of said comparison key recorded on said storage medium of said second device; and

programming said second device to transmit, in the event of a correct comparison, a further portion of said

second identification or authentication key to said first device.

71. The method of claim 70, further comprising programming said first and second devices to repeatedly transmit and compare portions of an identification or authentication key, until either all portions of said key have been transmitted, or an incorrect portion is transmitted.

72. The method of claim 12, further comprising

said keys being identification or authentication keys;

programming said user device or said host to transmit a checksum number related to a selected plurality of keys to said host device or user device, respectively.

73. The method of claim 72 wherein said selected plurality of keys is selected according to a predetermined algorithm.

74. The method of claim 12, further comprising

said keys including encryption keys;

establishing a communication session between said user device and said host; and

encrypting data transmitted between said host and said user device using as least one of said keys.

75. The method of claim 74, wherein an encryption key used to encrypt data transmitted between said host and said user device is padded with null or nonsense prefixes or suffixes chosen according to a predetermined algorithm.

76. The method of claim 74, wherein said set of user keys is a one time pad, each key being used to encrypt a partial sequence of said data.

77. The method of claim 74, wherein said user keys are used once.

78. The method of claim 74 wherein a key of said set of user keys is used to encrypt a limited or partial sequence of data in said communication session.

79. The method of claim 74, wherein a relative strength of an encryption level of said communication session is changed during said communication session.

80. The method of claim 12, wherein

a plurality of sets of user keys is recorded on said storage medium;

a first set of user keys provided to said user being associated with a predetermined resource; and

said user device not permitting access to said sets of user keys other than said first set thereof absent authorization from said host.

81. The method of claim 12 further comprising a portion of said resource being stored on said user device.

82. The method of claim 12, further comprising preforming a transaction with said user device, and storing information associated with said transaction on said user device.

83. The method of claim 82, wherein said transaction information stored on said user device is transmitted to said host.

84. The method of claim 16, further comprising

establishing a communication session between said user device and said host;

conducting a transaction process in parallel with and at the same time as an identification or authentication process.

85. The method of claim 84, wherein

during said identification or authentication process, said user device transmits a first and a second identification or authentication key to said host;

said host performs a relatively rapid confirmation of said first key;

upon confirmation of said first key, said host allows said transaction process to commence;

said host performs a confirmation of said second key;

said host terminating said transaction process if said second key is not confirmed.

86. The method of claim 85, wherein said first key is an encrypted form of said second key, said second key being encrypted according to a predetermined algorithm.

87. The method of claim 86, wherein said first and second keys are stored on said storage medium.

88. The method of claim 87, wherein said user device does not contain said predetermined encryption algorithm.

89. The method of claim 85, wherein pending confirmation of said second key, said host grants provisional limited access to said resource to said one user device.

90. The method of claim 12, wherein said user device is a portable electronic device having a memory and a processor, said set of user keys being recorded on said memory.

91. The method of claim 90 wherein said set of user keys recorded on said portable electronic device comprises an ultra-long database of keys, which keys are 25 characters or greater in size.

92. The method of claim 90, further comprising

said keys including identification or authentication keys; and

said portable electronic device including means to transmit said identification or authentication keys to said host.

93. The method of claim 90, further comprising

said keys including encryption keys; and

said portable electronic device including means to encrypt and decrypt data using said encryption keys and to transmit and receive encrypted data to and from said host.

94. The method of claim 90, wherein said portable electronic device includes means to prevent unauthorized access to said set of user keys recorded thereon.

95. The method of claim 12, wherein said user device includes means to prevent unauthorized access to said set of user keys recorded thereon.

96. The method of claim 94, wherein said means to prevent unauthorized access to said keys recorded on said portable electronic device comprises means to disable said portable electronic device after a predetermined number of unauthorized attempts to access said user keys.

97. The method of claim 94, wherein said means to prevent unauthorized access to said keys recorded on said portable electronic device comprises means to disable said portable

electronic device if keys are requested faster than a predetermined rate.

98. The method of claim 97, wherein said predetermined rate is based on a maximum transmission speed between said portable electronic device and said host.

99. The method of claim 94, wherein:

said portable electronic device is distributed to said authorized user after said set of user keys is recorded thereon; and

said method further comprises activating means to prevent access to said set of user keys stored on said portable electronic device during a time when said portable electronic device is being distributed to said authorized user.

100. The method of claim 99, wherein said portable electronic device includes a clock and wherein said portable electronic device deactivates said means to prevent access to said keys at a predetermined time and date.

101. The method of claim 99, wherein said means to prevent access to said set of user keys is deactivated only after said portable electronic device establishes a communication session with an activation host and said activation host transmits a deactivation key to said portable electronic device.

102. The method of claim 90, wherein said portable electronic device includes means to control the rate of use of the keys of said set of user keys stored thereon.

103. The method of claim 90, wherein said portable electronic device includes means to monitor usage of keys recorded thereon.

104. The method of claim 103, wherein said means to monitor usage of keys recorded on said portable electronic device comprises writable memory and programming to track usage of or to erase used keys.

105. The method of claim 12 further comprising programming said host or said user device to detect an unauthorized attempt to access said host, resource or user device by an unauthorized user, to transmit a marker to said unauthorized user, and to record said marker on a system used by said unauthorized user.

106. The method of claim 105, wherein said marker includes information about said unauthorized user or information unique to said unauthorized attempt to access said host, resource or user device.

107. The method of claim 105, further comprising said host or said user device includes programming to detect markers on other devices.

108. The method of claim 105, further comprising storing markers on a plurality of systems used by said unauthorized attacker during the unauthorized attempt to access said host, resource or user device whereby a route used by said unauthorized user can be retraced.

109. The method of claim 105, further comprising,

upon the detection of an unauthorized attempt to access said host, resource or user device, said unauthorized user is connected to a substitute host, which

substitute host establishes a simulated communication session with said unauthorized user by simulating access to said host, resource or user device to which said unauthorized user attempted to gain access; and

steps to identify and locate said unauthorized user during said simulated communication session.

110. The method of claim 16, further comprising

establishing a communication session between said user device and said host to conduct a transaction;

said host or said user device transmitting a second identification or authentication key to said user device or said host, respectively, during or upon termination of said communication session.

111. The method of claim 110, wherein said host or said user device transmits a plurality of identification or authentication keys to said user device or said host, respectively during a communication session, said transmissions being made at predetermined points in said communication session.

112. The method of claim 110, wherein said second key is transmitted after said transaction is consummated and wherein said transaction is held in abeyance until said second identification or authentication key is verified.

113. The method of claim 110, wherein said second identification or authentication key is transmitted by said user device or said host upon request of said host or said user device, respectively.

114. The method of claim 110, further comprising transmitting a check function related to data transmitted prior to said transmission of said second identification or authen-

tication key, whereby the integrity of the data transmitted between authentication events can be verified.

115. The method of claim 12, further comprising

said keys including encryption keys;

transmitting data between said user device and said host, said transmitted data representing static images or video images;

locations, orientations, chominance or luminance of individual pixels or groups of pixels of said static or video images being altered according to said encryption keys prior to transmission of said data.

116. The method of claim 115, wherein said set of user keys is a one time pad, each key of said set of user keys being used once to encrypt a predetermined amount of data.

117. The method of claim 16, wherein said user device is a telecommunication device.

118. The method of claim 12, further comprising

said keys including encryption keys;

said user device being a telecommunication device;

transmitting data between said telecommunication device and said host;

values or time locations of said data being altered according to said encryption keys prior to transmission of said data.

119. The method of claim 118, wherein said values of said transmitted data is altered by adding random noise to said data, said random noise being related to said encryption keys.

120. The method of claim 118, wherein said values of said transmitted data is altered in the frequency domain, said alterations being related to said encryption keys.

121. A method to control access to a resource accessible from a plurality of communication devices, comprising the steps of:

generating a plurality of sets of unique or class specific keys, said keys being ultra-long and/or said sets of keys being ultra-large;

programming first and second user devices to permit connection to said host device and to transmit or receive keys to or from said host;

programming said host device to permit connection to said first and second user devices and to transmit or receive keys to or from said first and second user devices;

distributing first and second sets of user keys to first and second authorized users, each set of user keys comprising one of said sets of unique or class specific keys;

said first and second sets of user keys being recorded on first and second storage media, respectively, said first and second storage media being integral to or compatible with said first and second user devices, respectively;

establishing a communication session between said first and second user devices through said host; and

encrypting data transmitted by said first and second user devices using said first and second sets of user keys, respectively.

122. The method of claim 121, further comprising:

creating a key database comprising a compilation of said first and second sets of user keys;

providing said host device with said key database; and

programming said host to decrypt data transmitted from said first and second user device using said first or second set user keys, respectively, and to encrypt said decrypted data using said second or first set of user keys, respectively.

123. The method of claim 121, further comprising:

creating a translation key database comprising a set of keys suitable to translate data encrypted with said first or second set of keys directly to data encrypted with said second or first set of keys, respectively;

recording said translation key database on a storage medium connected to said host; and

programming said host to translate data transmitted from said first or second communication device using said translation key database.

124. The method of claim 123 wherein said data is translated directly between an encryption for said first or second user device into an encryption for said second or first user device, respectively, without an unencrypted version of said data existing on said host.

125. The method of claim 121 further comprising:

programming said host to create a translation key suitable to translate data encrypted with a key from said first or second set of user keys directly to data encrypted with a key from said second or first set of user keys, respectively;

programming said host to translate data transmitted from said first or second communication device using said translation key.

126. The method of claim 125 wherein said translation key is created during or immediately prior to said communication session.

127. The method of claim 121 further comprising:

programming said host with a translation function to translate data encrypted with a key from said set of user keys of said first user device with a key from said set of user keys of said second or first user device, respectively

said translation function using a key from each of said sets of user keys of said first and second user devices.

128. A method to control access to a resource accessible from a plurality of communication devices, comprising the steps of:

generating a plurality of sets of unique or class specific keys, said keys being ultra-long and/or said sets of keys being ultra-large;

programming first and second user devices to permit connection to said host device and to transmit or receive individual or class specific keys to or from said host;

programming said host device to permit connection to said first and second user devices and to transmit or receive individual or class specific keys to or from said first and second user devices;

establishing a communication session between said first and second user devices;

said host transmitting a common set of keys to said first and second communication devices, said common set of keys comprising a set of said unique or class specific keys; and

encrypting data transmitted by said first and second user devices using said common set keys.

129. The method of claim 128, further comprising:

recording, on a first user device, a first set of user keys comprising a set of said individual or class specific keys;

recording, on a second user device, a second set of user keys comprising a set of said individual or class specific keys;

said common set of keys is encrypted using said first and second sets of user keys prior to transmission to said first and second user devices, respectively.

130. The method of claim 129 wherein said first and second user devices are distributed to said first and second users, respectively.

131. A portable electronic device, comprising:

a storage medium

a processor

a database of unique or class specific keys, said database being ultra-large or said keys being ultra-long; and

means to connect said portable electronic device to an external resource.

132. A portable electronic device as in claim 131, wherein said keys are at least one 25 characters or greater in length.

133. A portable electronic device as in claim 131, further comprising first and second means to connect to first and second external resources, respectively.

134. A portable electronic device as in claim 133, further comprising first and second processors, said first and

second processors being connected to said first and second means to connect to first and second external resources, respectively.

135. A portable electronic device as in claim 134, wherein said first means to connect to an external resources further comprises means to interface with a media reader of a first type and wherein said second means to connect to an external resources further comprises means to interface with a media reader of a second type, different than said first type of media reader.

136. A portable electronic device as in claim 134, wherein said first and second processors are connected to said storage medium.

137. A portable electronic device as in claim 133, wherein said first and second means to connect to said first and second external resources further comprises first and second electronic circuits.

138. A portable electronic device as in claim 133, wherein said first and second means to connect to said first and second external resources further comprises first and second databases recorded on said storage medium.

139. A portable electronic device as in claim 133, wherein said first and second means to connect to said first and second external resources further comprises first and second storage media.

140. A portable electronic device as in claim 133, wherein said first and second means to connect to said first and second external resources further comprises said portable

electronic device being programmed with first and second identification or authentication algorithms.

141. A portable electronic device as in claim 133, wherein said first and second means to connect to said first and second external resources further comprises said portable electronic device being programmed with first and second communication protocols.

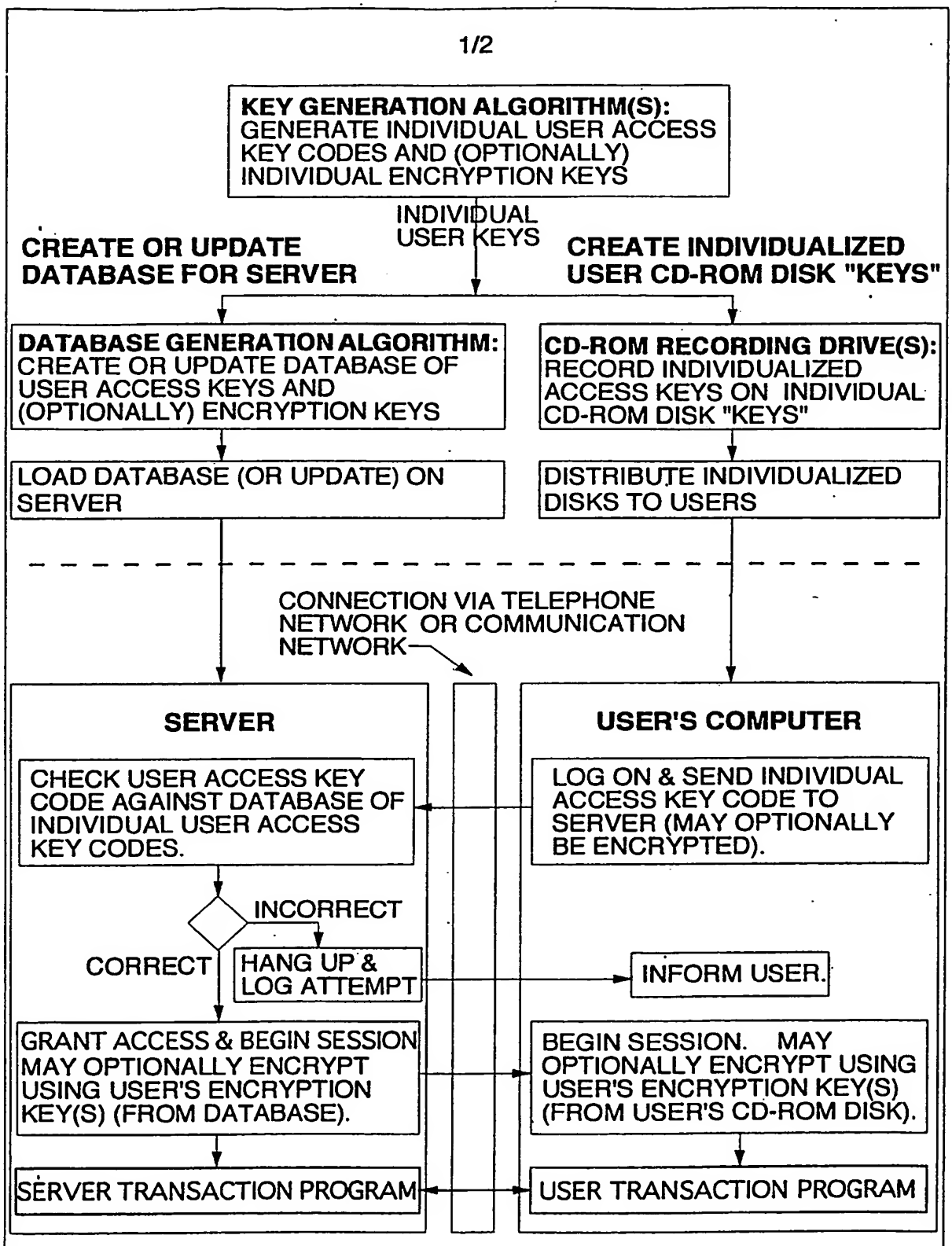
KEY GENERATION ALGORITHM(S): GENERATE INDIVIDUAL USER ACCESS KEY CODES AND (OPTIONALLY) INDIVIDUAL ENCRYPTION KEYS
INDIVIDUAL USER KEYS
CREATE OR UPDATE DATABASE FOR SERVER
CREATE INDIVIDUALIZED USER CD-ROM DISK "KEYS"
DATABASE GENERATION ALGORITHM: CREATE OR UPDATE DATABASE OF USER ACCESS KEYS AND (OPTIONALLY) ENCRYPTION KEYS
CD-ROM RECORDING DRIVE(S): RECORD INDIVIDUALIZED ACCESS KEYS ON INDIVIDUAL CD-ROM DISK "KEYS"
LOAD DATABASE (OR UPDATE) ON SERVER
DISTRIBUTE INDIVIDUALIZED DISKS TO USERS
CONNECTION VIA TELEPHONE NETWORK OR COMMUNICATION NETWORK
SERVER
USER'S COMPUTER
CHECK USER ACCESS KEY CODE AGAINST DATABASE OF INDIVIDUAL USER ACCESS KEY CODES.
LOG ON & SEND INDIVIDUAL ACCESS KEY CODE TO SERVER (MAY OPTIONALLY BE ENCRYPTED).
INCORRECT
CORRECT
HANG UP & LOG ATTEMPT
INFORM USER.
GRANT ACCESS & BEGIN SESSION MAY OPTIONALLY ENCRYPT USING USER'S ENCRYPTION KEY(S) (FROM DATABASE).
BEGIN SESSION. MAY OPTIONALLY ENCRYPT USING USER'S ENCRYPTION KEY(S) (FROM USER'S CD-ROM DISK).
SERVER TRANSACTION PROGRAM
USER TRANSACTION PROGRAM

FIG. 1

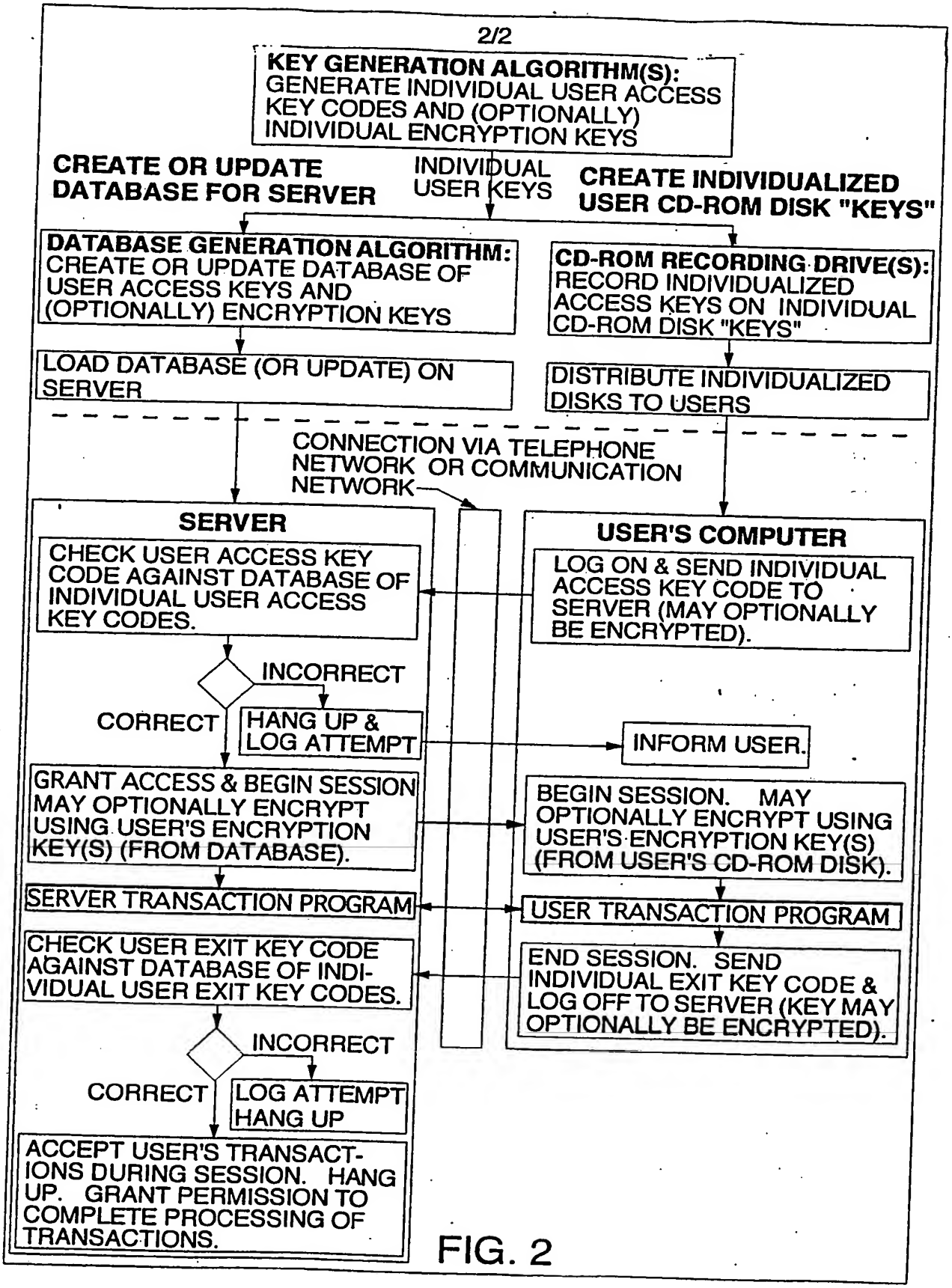
KEY GENERATION ALGORITHM(S): GENERATE INDIVIDUAL USER ACCESS KEY CODES AND (OPTIONALLY) INDIVIDUAL ENCRYPTION KEYS
CREATE OR UPDATE DATABASE FOR SERVER
INDIVIDUAL USER KEYS
CREATE INDIVIDUALIZED USER CD-ROM DISK "KEYS"
DATABASE GENERATION ALGORITHM: CREATE OR UPDATE DATABASE OF USER ACCESS KEYS AND (OPTIONALLY) ENCRYPTION KEYS
CD-ROM RECORDING DRIVE(S): RECORD INDIVIDUALIZED ACCESS KEYS ON INDIVIDUAL CD-ROM DISK "KEYS"
LOAD DATABASE (OR UPDATE) ON SERVER
DISTRIBUTE INDIVIDUALIZED DISKS TO USERS
CONNECTION VIA TELEPHONE NETWORK OR COMMUNICATION NETWORK
SERVER
USER'S COMPUTER
CHECK USER ACCESS KEY CODE AGAINST DATABASE OF INDIVIDUAL USER ACCESS KEY CODES.
LOG ON & SEND INDIVIDUAL ACCESS KEY CODE TO SERVER (MAY OPTIONALLY BE ENCRYPTED).
INCORRECT
CORRECT
HANG UP & LOG ATTEMPT
INFORM USER.
GRANT ACCESS & BEGIN SESSION MAY OPTIONALLY ENCRYPT USING USER'S ENCRYPTION KEY(S) (FROM DATABASE).
BEGIN SESSION. MAY OPTIONALLY ENCRYPT USING USER'S ENCRYPTION KEY(S) (FROM USER'S CD-ROM DISK).
SERVER TRANSACTION PROGRAM
USER TRANSACTION PROGRAM
CHECK USER EXIT KEY CODE AGAINST DATABASE OF INDIVIDUAL USER EXIT KEY CODES.
END SESSION. SEND INDIVIDUAL EXIT KEY CODE & LOG OFF TO SERVER (KEY MAY OPTIONALLY BE ENCRYPTED).
INCORRECT
CORRECT
LOG ATTEMPT HANG UP
ACCEPT USER'S TRANSACTIONS DURING SESSION. HANG UP. GRANT PERMISSION TO COMPLETE PROCESSING OF TRANSACTIONS.

FIG. 2

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/07174

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) :Please See Extra Sheet.
US CL :713/155, 200; 380/277

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 713/155, 183, 185, 200, 202; 380/277, 282, 285

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

IBM Technical Disclosure Bulletins

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS search terms: one time pad, authentication, multilevel security, access control, key management.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 4,605,820 A (CAMPBELL, JR.) 12 AUGUST 1986, col. 4, lines 10-14; col. 3, lines 18-67; col. 7, lines 52-65.	12-17, 21-23, 28-30, 41-43, 46, 131-133
----		-------
Y		1-11, 18-20, 24-27, 31-40, 44-45, 47-120, 121-127
Y	US 4,145,568 A (EHRAT) 20 MARCH 1979, Fig. 1--#10, 11; col. 2, lines 44-55.	24-25, 31-35, 49
Y	US 5,048,085 A (ABRAHAM et al) 10 SEPTEMBER 1991, Fig. 8; col. 9, lines 5-33.	18-20, 26-27, 36-39, 79, 97-102

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 02 JUNE 2000	Date of mailing of the international search report 07 JUL 2000
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer GILBERTO BARRÓN James R. Matthews Telephone No. (703) 305-3800/4700

Form PCT/ISA/210 (second sheet) (July 1998)*

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/07174

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,272,754 A (BOERBERT) 21 DECEMBER 1993, Fig. 4, #118, col. 10, lines 1-11	2, 40, 96
Y	US 4,731,841 A (ROSEN et al) 15 MARCH 1988, Fig. 2, col. 6, lines 6-30	50-95
Y	US 5,293,576 A (MIHM, JR. et al) 08 MARCH 1994, Fig. 6B, col. 15, lines 47-60.	128-130
Y	US 5,261,070 A (OHTA) 09 NOVEMBER 1993, col. 3, lines 44-60.	3-6, 8-11, 131-133
Y	US 4,960,982 A (TAKAHIRA) 02 OCTOBER 1990, Fig. 2, col. 4, lines 46-60.	134-141

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/07174

A. CLASSIFICATION OF SUBJECT MATTER:
IPC (7):

H04L 9/32